TABLE OF CONTENTS
A1307-300-SSS-2
28 August 1997

SYSTEM SPECIFICATION FOR THE THREAT SIMULATOR LINKING ACTIVITIES NETWORK

CONTRACT NO. F08635-92-C-0050
CDRL SEQUENCE NO. (A025)

Prepared for:
Joint Advanced Distributed Simulation Program Office
Joint Test Force
Kirtland AFB, NM 87117

Prepared by:
Georgia Institute of Technology
Georgia Tech Research Institute
Atlanta, Georgia 30332

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. SCOPE

1.1. IDENTIFICATION

Linking of existing test facilities is seen as a method for enhancing the realism and thoroughness of Test and Evaluation (T&E) for Electronic Warfare (EW) assets. These individual test facilities, for the most part, already exist. The linking is to be accomplished through the communication services available from the commercial long-haul carriers which also already exist. The network interactions are to be carried out according to the interface specifications defined by the Defense Modeling and Simulation Office in the High Level Architecture Interface Specification.

This System Specification identifies the entity/facility types, communication networks, and the High Level Architecture (HLA) which will support the implementation of an Advanced Distributed Simulation (ADS) network as required to perform Test & Evaluation (T&E) of Electronic Warfare (EW) assets in realistic combat environments. For convenience, this overall network shall be referred to as the Threat Simulation Linking Activities (TSLA) Network.

Although classes of simulation entities are defined in this specification for completeness, this specification does not define requirements for the implementation of such entities within test facilities. The focus of this System Specification is on the capabilities needed to interface these entities and/or test facilities to an HLA-compliant network, and on the specific data collection, formatting and reduction processing needed to perform an integrated linked EW test. The class definitions provided in this specification are limited to systems which operate in the RF frequency range.

1.2. SYSTEM OVERVIEW

The *system* to be specified is actually a collection of assets networked together to form an integrated piece of test equipment. The assets include the existing facilities which support EW T&E, the commercial long-haul leased communication networks which support transport of test and control data, and the Run-Time Infrastructure (RTI) developed to provide HLA services.

1.3. DOCUMENT OVERVIEW

Requirements for the communication infrastructure and the RTI are incorporated here by reference. The coupling of these assets to the remaining system assets is addressed in this specification. Enumerated requirements appear in several sections of this document in addition to the section text. These enumerated requirements are to be considered in addition to the requirements defined in the text.

2. APPLICABLE DOCUMENTS

2.1. GOVERNMENT DOCUMENTS

The following documents of the exact issue shown form a part of this specification to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this specification, the contents of this specification shall be considered a superseding requirement.

2.1.1. SPECIFICATIONS

1. MIL-T-31000 - General specifications for Technical Data Package Lists
2. High Level Architecture Rules, version 1.0, dated 15 Aug 1996
3. High Level Architecture Interface Specification, version 1.1, dated 12 Feb 1997
4. High Level Architecture Object Model Template, version 1.1, dated 12 Feb 1997
5. Network Requirements Specification for Threat Simulator Linking Activities in support of EW Testing, 22 April 1997.

2.1.2. STANDARDS

1. MIL-STD 1521B, Notice 1 - Technical Reviews and Audits for Systems, Equipments, and Computer Software
2. MIL-STD-129K, Notice 1 - Marking for Shipment and Storage
3. MIL-STD 481B - Configuration control - engineering Changes (short form), Deviations and Waivers
4. MIL-STD-483A - Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs
5. MIL-STD-794E, Notice 1 - Parts and Equipment, Procedures for Packaging of,
6. MIL-STD-882B, Notice 1 - System Safety Program Requirements
7. MIL-STD-100E - Engineering Drawing Practices
8. MIL-STD-461C, Notice 2 - Electromagnetic Emission and Susceptibility Requirements for Control of Electromagnetic Interference
9. MIL-STD-462, Notice 6 - Electromagnetic Interference Characteristics, Measurements of
10. World Geographic Society Standard (1984) MIL-STD-2401 World Geodetic System, WGS-84.

2.1.3. OTHER

1. AF-MAN- 99-112: Air Force Electronic Warfare T&E Process Manual, 27 March 1995.
2. Feasibility Study Report, “Joint Advanced Distributed Simulation,” Prepared by ODDR&E (T&E), Final Report, February 1995.
3. DoD 5220.22M - Industrial Security Manual for Safeguarding Classified Information

2.1.4. DRAWINGS

None

2.1.5. OTHER PUBLICATIONS

None

2.2. NON-GOVERNMENT DOCUMENTS

The following documents of the exact issue shown form a part of this specification to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this specification, the contents of this specification shall be considered a superseding requirement.

2.2.1. SPECIFICATIONS

None

2.2.2. STANDARDS

1. Information technology--Local and metropolitan area networks--Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, ANSI/IEEE Std 802.3, 1996 Edition
2. Information technology--Telecommunications and information exchange between systems--Local and metropolitan area networks--Specific requirements--Part 2: Logical link control, ANSI/IEEE Std 802.2, 1994 Edition.
3. Synchronization Interface Standards for Digital Networks (ANSI T1.101-1994)
4. Digital Hierarchy - Electrical Interfaces (ANSI T1.102-1993)
5. Exchange-Interexchange Carrier Interfaces-Individual Channel Signaling Protocols (ANSI T1.104-1991)
6. Digital Hierarchy - Formats Specifications (ANSI T1.107-1995)
7. IEEE-STD-1278.1 - Standards for distributed Interactive Simulation (DIS) -- Application Protocols, 1993.
8. Current InterNIC standards on TCP/IP protocol and document in current Request For Comments (RFC) Documents. At the time of this specification, the current RFC was 1780.

2.2.3. DRAWINGS

None

2.2.4. OTHER PUBLICATIONS

None

3. SYSTEM REQUIREMENTS

3.1. SYSTEM DEFINITION

A functional representation of the overall TSLA Network is shown in Figure 1. Each of the general classes of entities is represented in the figure; however, there are several members in each class which are

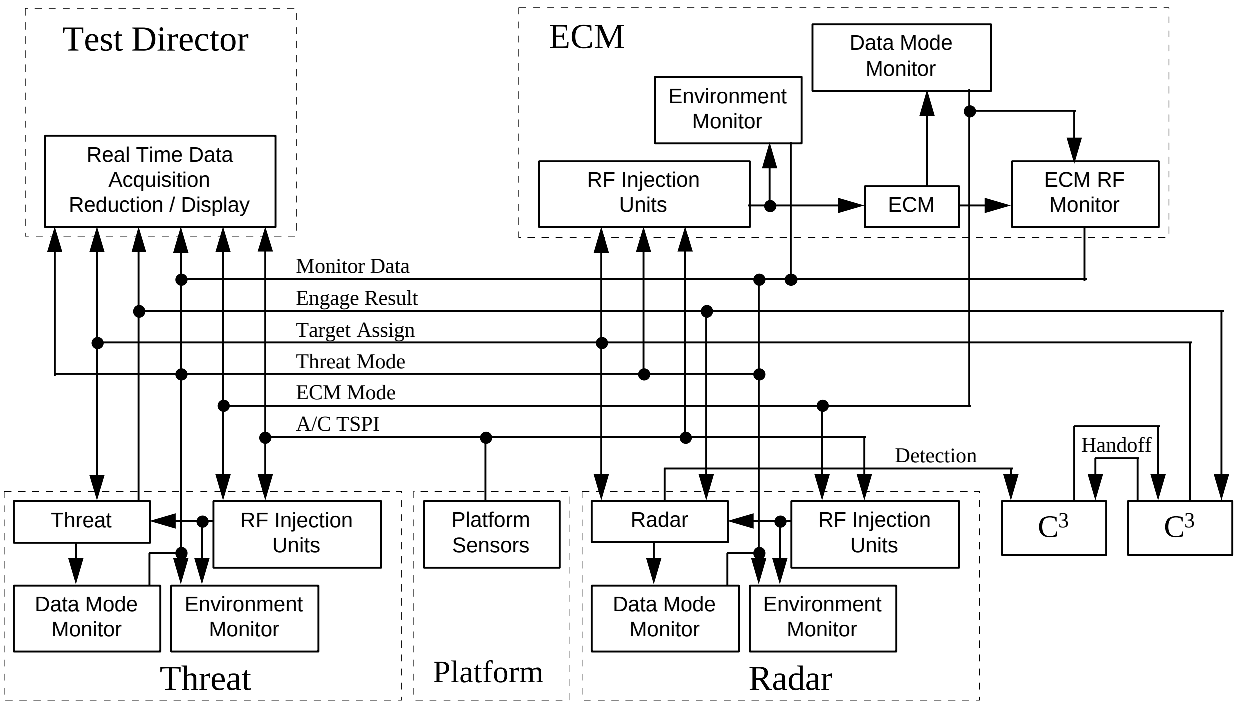

Figure 1. Overall Structure of the TSLA Network.

not explicitly represented. The complete membership of each class is shown in Figure 2 through Figure 5.

3.1.1. ENTITY CLASS DEFINITIONS

The classes represented in Figure 1 contain several members. The overall class structure tables are shown in Figure 2 through Figure 5.

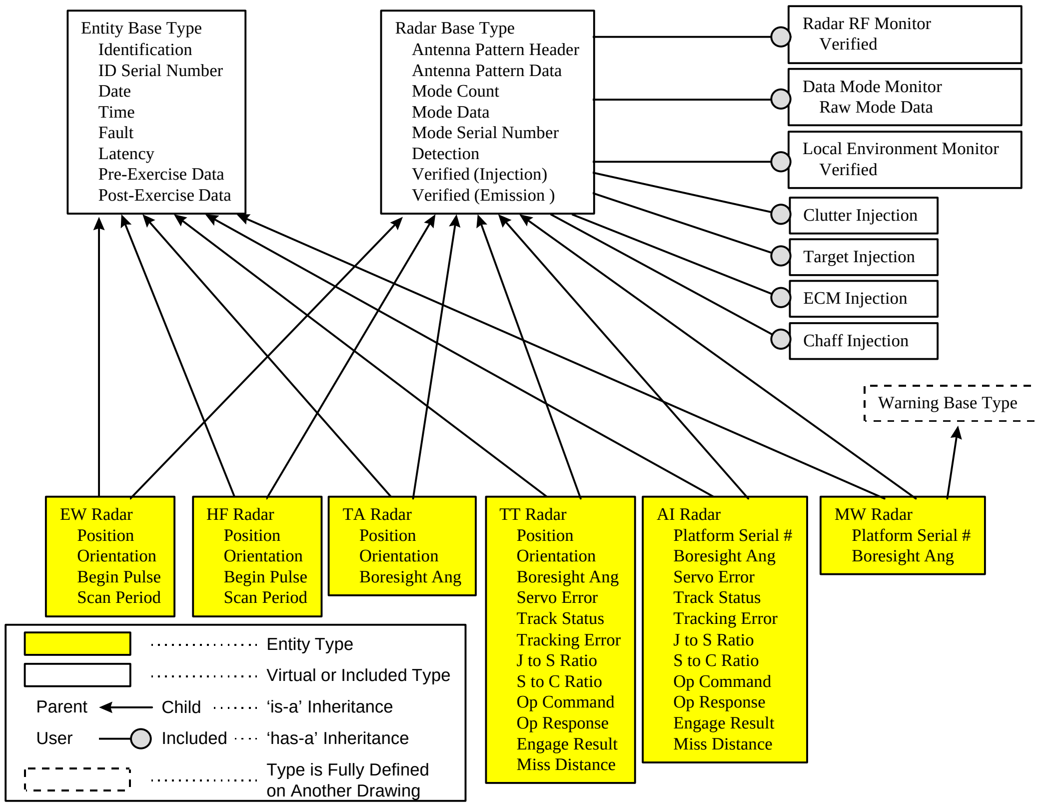

Figure 2. Object Oriented Hierarchy for Published Data, Radar Entities

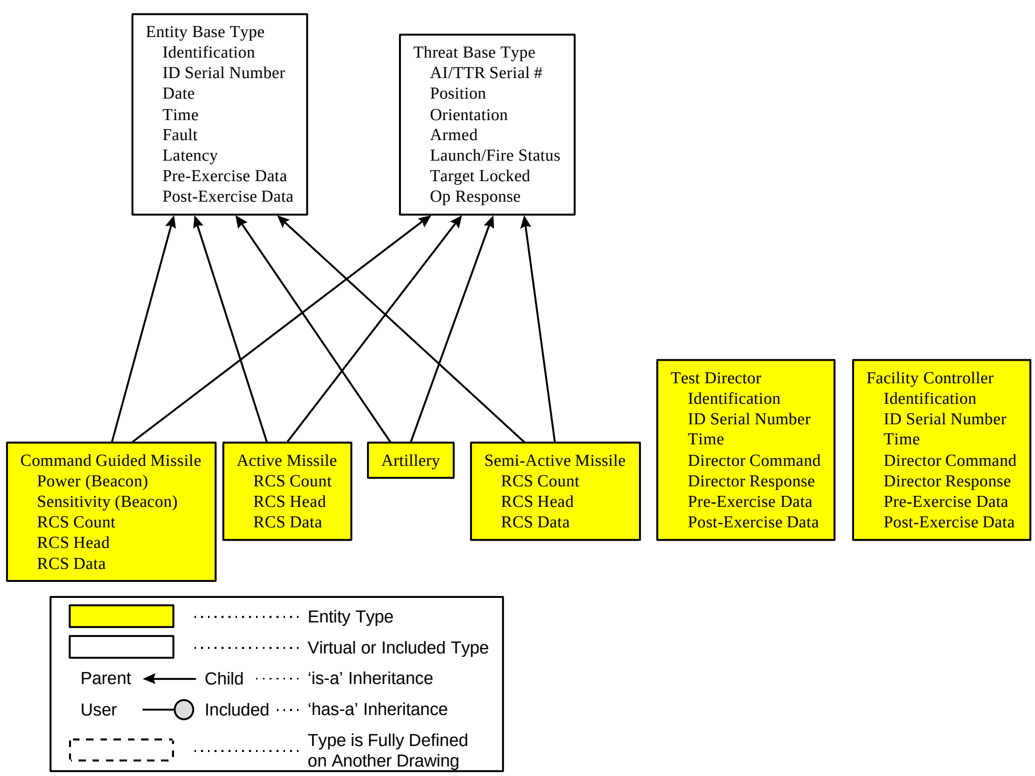

Figure 3. Object Oriented Hierarchy for Published Data, Threat and Control Entities

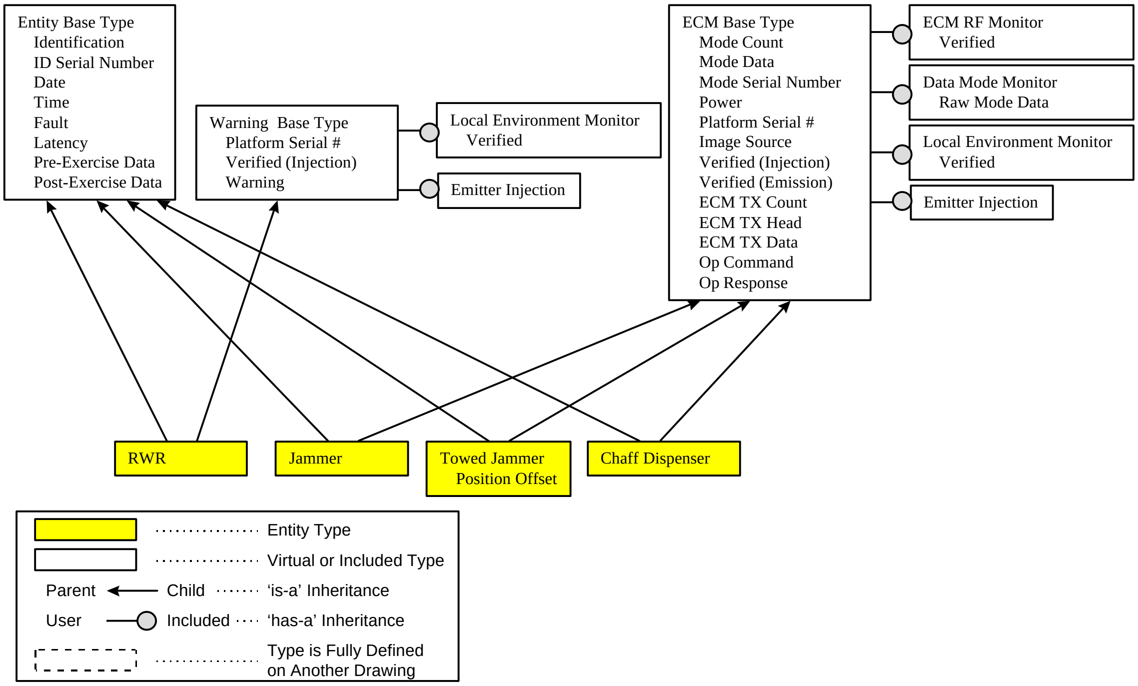

Figure 4. Object Oriented Hierarchy for Published Data, ECM Entities

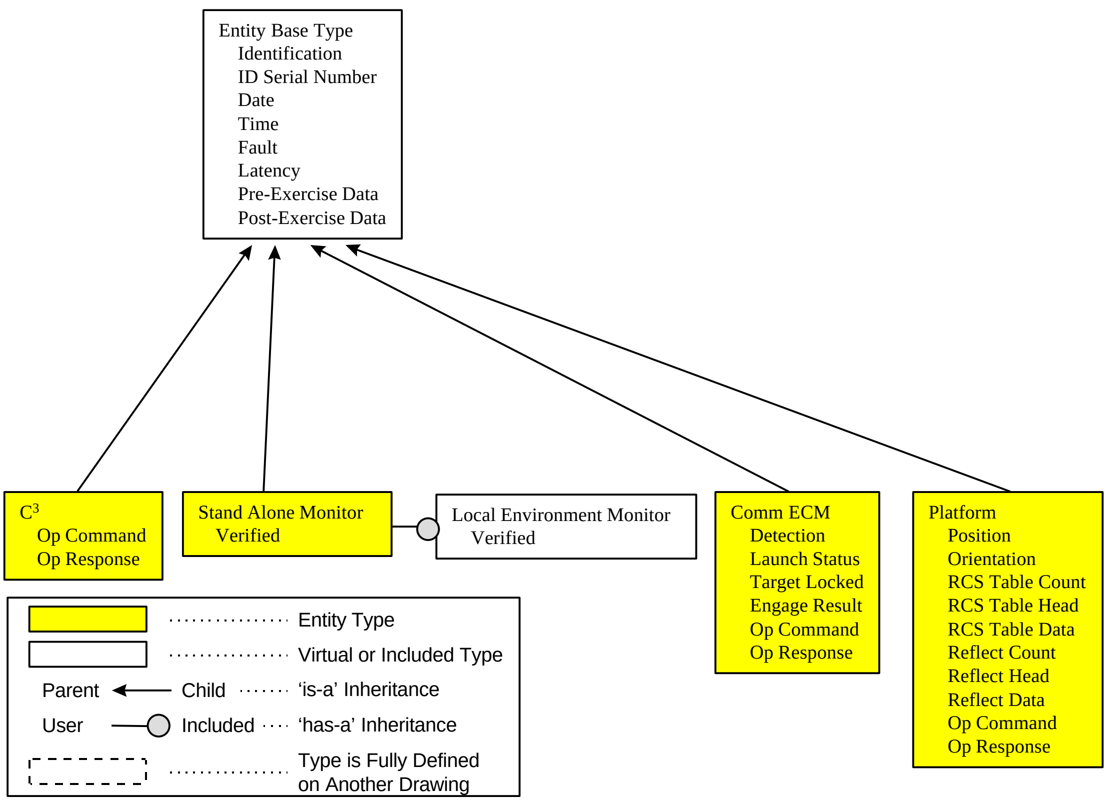

Figure 5. Object Oriented Hierarchy for Published data, Platform, C^3, Stand-alone Monitors, and Communication ECM

3.1.1.1. RADAR CLASS FUNCTIONAL DESCRIPTION

Radar entities are used to represent any radar-based system with collocated transmitter and receiver. The specific types of radar entities which are members of the Radar class include,

Radar

- *Early Warning Radar (EWR),*
- *Height Finder Radar (HFR),*
- *Target Tracking Radar (TTR),*
- *Target Acquisition Radar (TAR)*
- *Airborne Interceptor Radar (AIR)*
- *Missile Warning Radar (MWX).*

3.1.1.2. THREAT CLASS FUNCTIONAL DESCRIPTION

Threat entities are used to represent any system that may be used to disable or destroy a platform entity. The specific types of threat entities which are members of the Threat class include,

Threat

- *Active Missile*
- *Command Guided Missile*
- *Artillery*
- *Semi-active Missile*

Threats listed as missiles represent self-propelled, airborne ordnance with a capacity for guidance after launch. Artillery threats represent ordnance that follow a ballistic trajectory with no capacity for guidance after firing.

3.1.1.3. ECM CLASS FUNCTIONAL DESCRIPTION

ECM entities are used to represent any electronic countermeasure system that may be carried by a platform. The specific types of ECM entities which are members of the ECM class include,

ECM

- *Jammer*
- *Towed Jammer*
- *Radar Warning Receiver (RWR)*
- *Expendables*

An ECM entity receives incoming radiation from all sources and generates a response designed to defeat one or more threat systems. The jammer type of ECM entity responds with an RF emission. The RWR type of ECM entity responds with warning data. In either case, the response may be codified within the mode information published by an ECM entity.

The expendable entity is used to represent the behavior of a chaff or free-fall expendable dispenser. This entity reports the remaining expendable supply, confirms dispense in response to dispense commands, and provides expendable type and dispense information to the radar signal injection unit which provides the actual RF stimulation to represent the signals resulting from the expendables.

3.1.1.4. PLATFORM CLASS FUNCTIONAL DESCRIPTION

Data from the platform are involved in calculating three system-level parameters: skin return, ECM return, and ECM input. No single entity generates all the information needed to calculate any of these parameters. The platform contributes data to other entities in support of these calculations. In summary, the platform must provide RCS tables, along with position and attitude to radar entities. Associated ECM entities require position and attitude along with the transmit and receive antenna patterns. Although the ECM antenna patterns are similar in measure to the RCS tables, the ownership of the antenna patterns shall remain with the ECM entity. This choice was made owing to the manner in which data table maintenance duties are normally assigned.

3.1.1.5. C^3 CLASS FUNCTIONAL DESCRIPTION

Two basic types of C^3 nodes are defined for the TSLA Network: Tactical and Strategic. The tactical C^3 node receives reports from the target acquisition radar (TAR), or from higher echelon (Strategic) C^3 assets. The primary function of a tactical C^3 entity is to track, select, and assign targets for engagement. A tactical C^3 node assigns targets directly to the target engagement systems (TES). TES configurations generally include a transporter erector launcher and radars (TELAR), or a target tracking radar (TTR) accompanied by a separate transporter erector launcher (TEL). The tactical C^3 must also select fire units and assign specific targets to specific fire units. To aid in target assignment and target engagement functions, tactical C^3 systems generally establish and maintain track filters. The track filters allow the tactical C^3 entity to extrapolate target positions between target acquisition radar (TAR) updates and to assign higher priorities to the most threatening targets. Also in this role, the tactical C^3 entity may coordinate the activities of several engagement systems against a common target. In such engagements, the target may be tracked by one engagement system while a separate engagement system is used to launch the missiles and/or to provide target illumination for missile homing.

A secondary function of a tactical C^3 entity is to accept target designations from a higher level (e.g., strategic) C^3 system and perform data fusion on the designations. In some cases the tactical C^3 will receive its target information directly from a TAR. The TAR may be a separate entity or may be a component of an associated TES. In other cases the tactical C^3 node may receive its target information from another, higher echelon C^3 system. A target assignment from a higher level C^3 system can be correlated with returns from a separate TAR to determine if multiple targets are present and to avoid unintentionally assigning the same target to several different engagement systems.

The Ground-Controlled Intercept (GCI) Function is also considered to be a C^3 entity in the TSLA Network. In this case, the TAR is ground-based, but the TES is airborne. For the airborne interceptor, additional data is communicated between the GCI C^3 facility and the airborne TES to vector the TES platform to the target of interest.

3.1.1.6. STAND ALONE MONITOR CLASS FUNCTIONAL DESCRIPTION

Environment monitors are specified as part of the instrumentation suite for many of the entities addressed later in this specification. (See Section 3.7.) In addition to these monitors, it is necessary to include a monitor to perform general surveillance in vicinity of any open-air assets participating in the test. This monitor, instead of checking on the performance of any of the entities, will check that there are no extraneous emitters in the test environment. This monitor uses the same mode information from emitting entities as that used by the entity monitors. However, the Stand-Alone Environment Monitor uses this information to determine what to eliminate from its signals of interest. If, after eliminating all of the signals reported by legitimate entities in the test, there remain other signals, it is the job of the Stand-Alone Environment Monitor to make this determination and report this information to the Test Director Facility.

3.1.1.7. COMM ECM CLASS FUNCTIONAL DESCRIPTION

The Communication ECM entity intercepts data, potentially alters the data, and republishes the data to the original subscriber. Not all data are subject to interception by the Communication ECM entity. Command data, telemetry data, and other such data that might be communicated between entities during a live exercise are possible targets for interception. Data that exist solely due to the distributed nature of the EW test (e.g., radar mode data) are not targets for communication ECM.

3.1.1.8. FACILITY CONTROLLER CLASS FUNCTIONAL DESCRIPTION

Many test facilities are already organized around a common network gateway that explicitly supports the use of a facility as the physical point of connection to the network and as a logical control point for the entities within its purview. The Facility Controller is the abstraction that allows all entities to properly respond to commands from the Test Director Facility. The Facility Controller may also abstract certain elements of each entity into its own function.

3.1.1.9. TEST DIRECTOR CLASS FUNCTIONAL DESCRIPTION

The Test Director Facility is responsible for monitoring and controlling all test activities during a test. The Test Director Facility shall monitor,

1. data necessary to analyze the quality of the test, and
2. data necessary to evaluate the performance and/or effectiveness of the System Under Test (SUT).

These data shall include at a minimum:

1) scenario visualization data such as:

scenario map,
threat laydown,
target position versus time, and
threat mode versus times;

2) QA data, including:

SUT response versus time,

filter center plot board,
verified emitter mode versus time,
verified SUT response versus time, and
extraneous emitters vs time;

3) network status data, including:

the result of BIT procedures,
the result of entity latency measurement, and
checksum error/noisy data reports;

4) SUT Modes,
5) threat track files,
6) ECM tables,
7) blanking statistics,
8) output power characteristics, 9) others as required to support RWR, MWR, towed decoy jammer entities,
10) perceived target position
11) Jamming/Signal Ratio, and
12) Attrition.

The Test Director Facility shall control the test execution by

1. Setting up the federation execution,
2. providing measurement scripts to instrumentation,
3. configuring the network,
4. transmitting commands to the affected nodes, and
5. terminating the federation execution.

To automate certain measurements, the Test Director Facility shall supply measurement scripts to particular instrumentation systems. The Test Director Facility shall be responsible for initializing network nodes and maintaining the status of the network. Commands processed by the Test Director Facility include controlling the test execution (e.g., Start test, Stop test, Pause test, and Initialize test), and reconfiguring Test Director displays (e.g., type of data to display, and size of a display window).

3.2. CHARACTERISTICS

3.2.1. PERFORMANCE CHARACTERISTICS

The TSLA network shall operate as an HLA federation. There are several activities which must take place in order to bring the federation to a point where a federation execution may be initiated. The activities required to *construct* the federation are not considered actions or *modes* of the federation execution. Strictly speaking, a federation cannot be placed in a particular mode, but rather each entity participating in a federation execution may be placed in a particular mode. Thus, some latitude in terminology is taken in this discussion of the modes of the federation execution. The modes are actually applied to the entities; however, the entire federation cannot move on to the next mode until all entities have moved on to the next mode.

3.2.1.1. SYSTEM OPERATION

The TSLA Network will function as a complex piece of integrated test equipment. As with any test equipment it is necessary and prudent to check out this equipment prior to using it to measure the performance of the system to be tested. The process of checking out the TSLA Network is defined in Table 1. This particular process is meant to confirm the readiness of the TSLA Network to perform the required test and to do so in a manner which is consistent with the requirements of HLA.

Table 1. TSLA Network Management Procedure.

PROCEDURE	NETWORK STATE	ACTION
Facility Setup and Self-Test	Pre-federation	Individual facility managers perform internal self-test and calibration
Confirm Ready to Join	Pre-federation	Test Director contacts each facility manager by voice to confirm readiness to join the federation.
Federation Create	Federation/Initialize	The Test Director sends commands to the local RTI to cause the creation of a federation. Also at this time, the Test Director initiates publication of any attributes to be published by the Test Director Facility.
Federates Join	Federation/Initialize	Test Director commands all federates to join federation. Individual federates respond by registering all publication and subscription requirements with the RTI.
Roll Call	Federation/Initialize	Upon request from the Test Director Facility, each federate will respond through the network to indicate readiness. The readiness response shall contain a report of: FRED include rev RTI include rev API include rev
Standby	Federation/Standby	After a federate has responded successfully to the roll call, the Test Director Facility shall issue a message to that federate to enter standby condition.
Pub-Sub connectivity Connectivity Confirmed	Federation/Execution Federation/Standby	A special scenario, included at the beginning of the test procedure will be executed for the purpose of confirming that all subscribers have a good connection to the required publisher with tolerable latency. As each federate verifies its data connections, it reports to the Test Director Facility that connectivity is confirmed. When the Test Director Facility obtains this report, it instructs the federate to go back to Standby condition.
Test Execution	Federation/Execute	The actual test proceeds on command to start which is broadcast from the Test Director Facility.

PROCEDURE	NETWORK STATE	ACTION
Post Test Roll Call	Federation/Post-test	Following the test, each facility manager shall perform internal self-test and calibration. When polled by roll call, each federate shall report test readiness condition.
Log Data Transfer	Post-federation	Any federate data which is required at any other location is transported by FTP, email, CD-R, etc.

3.2.1.2. STATE 1: INITIALIZATION MODE

Req. 1. A Roll call mode shall be supported by each entity to permit the Test Director to query the state of readiness of each entity prior to beginning the test.

Req. 2. A Standby mode shall be supported by each entity to permit the Test Director to hold selected entities ready to begin at the chosen start time.

The initial operation after the creation of the normal federation execution is the entity *roll call*. As each entity reports, the Test Director will examine the entity's status to determine whether the entity is prepared to proceed with the test. Each entity shall be provided five seconds to respond to the roll call. If a response is not received within this time period, the entity shall be considered unavailable to the federation.

When the readiness status of an entity is confirmed, the Test Director shall issue a command that will allow that entity to enter *standby condition*.

3.2.1.3. STATE 2: EXECUTE MODE

Req. 3. An Execute mode shall be provided by each entity to permit the entity to interact with the other entities in the federation according to the Federation Object Model.

When the Test Director is satisfied that the federation is ready to proceed with the test, the Test Director shall issue a command which places the federation in *execute mode*. During execute mode, the Test Director shall have the capability to pause the test and replay portions of the test or restart the test. If live entities are participating, a replay or restart shall cause any constructive entities to pause while the live and virtual entities return to the correct replay or restart state.

Certain maintenance functions may best be performed in the context of a federation (e.g., testing for HLA compliance and overall network reliability, latency, and connectivity) This type of testing and diagnostic data collection shall be performed within the Execute Mode. The user will define a test scenario suitable for the test of interest. Specific data collection shall include the QA data described in Section 3.1.1.9 at a minimum. Additional data requirements shall be assessed in preparation of the lower level development specifications.

3.2.1.4. STATE 3: POST TEST MODE

Req. 4. A Post-test mode shall be provided by each entity to permit the Test Director to query each entity for any required test details or post-test diagnostic information.

At the conclusion of the test, before the entities resign from the federation and before the federation execution ceases to exist, the Test Director shall issue a command to initiate *post-test mode*. Post-test mode is executed in a similar fashion to roll call mode. Each entity is requested to provide a post-test report of readiness condition.

After the conclusion of the federation execution, if detailed post-test data (e.g., log data) are required at a location other than the location where they were generated, the data shall be transmitted by normal methods (e.g., FTP, floppy disk, CD-R, etc.). Defining the specific contents and format of the data logs for post-test analysis is part of the federation development process.

3.2.2. SYSTEM CAPABILITY RELATIONSHIPS

The overall network is comprised of several components. Many of these components already exist or are in various stages of development. In particular, there are several test facilities which offer existing capabilities for several of the required entity classes. An overall network architecture is shown in Figure 6 to standardize the development of the TSLA Network. This section and its subsections provide specifications for the system segments needed for these entity classes to interface to the TSLA Network. Instrumentation requirements needed to transform between the native signal domain of the entities and the data domain of the network are specified in subsections of 3.7.

Each of the entities/facilities shall employ common capabilities to interface to the network. A Facility Control service, where it is currently employed by an entity/entity facility, may remain in use at the facility which supports the entity(ies). The HLA Network Interface is the software layer which converts the interface provided by the entity/entity Facility Control to that required by HLA. The Physical Network Interface is the low level driver interface needed to support the connection to the physical transport layer used in the TSLA Network.

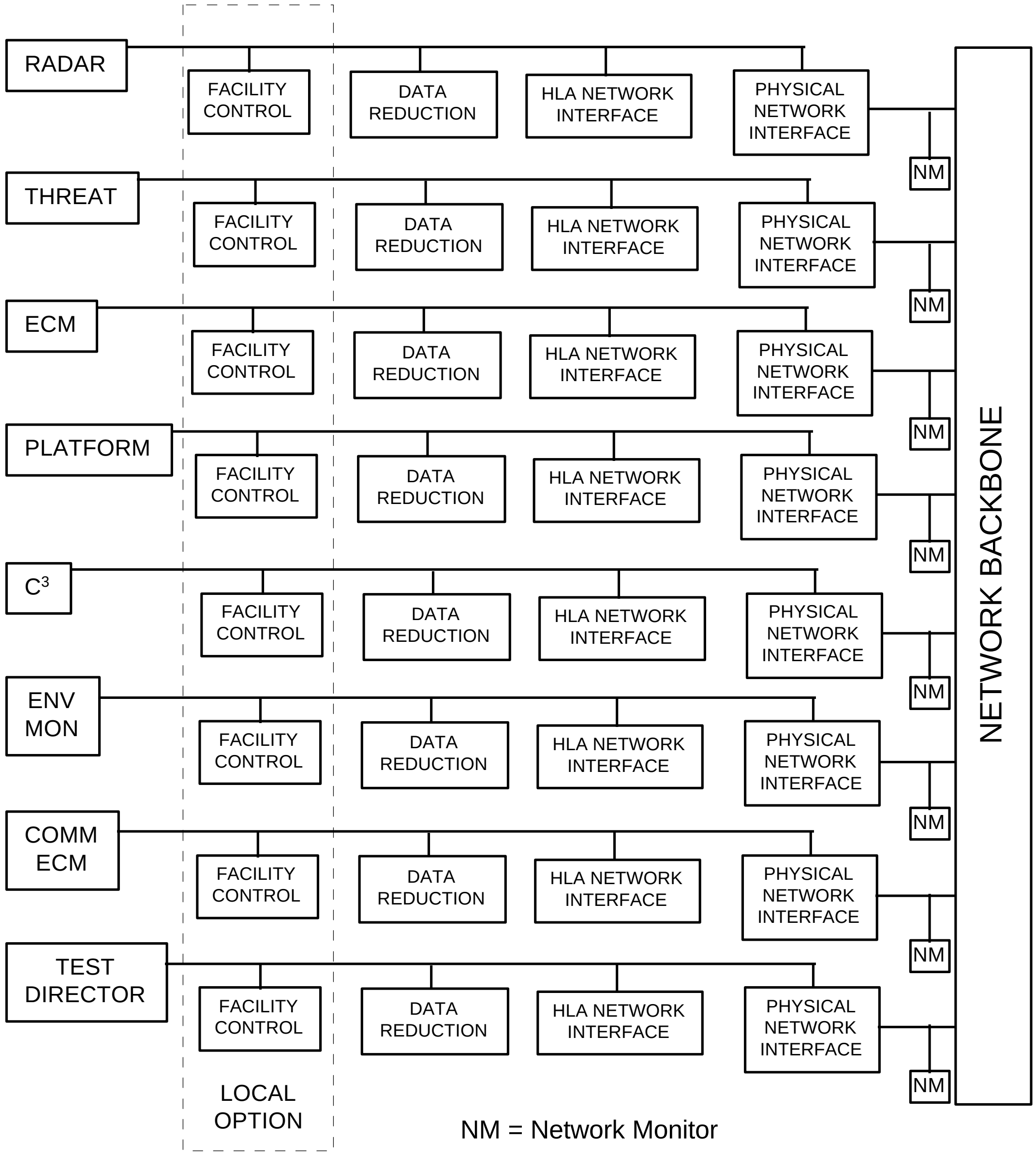

Figure 6. TSLA Network Interface Architecture.

3.2.3. System Segments

3.2.3.1. Facility Control

Facility Control is that function which serves as the gateway to a multi-entity facility. The functions performed by this system segment are not delineated in this specification. Preservation of the existing Facility Controller function is considered to be in compliance with this specification.

3.2.3.2. HLA NETWORK INTERFACE

To interface to the TSLA network, this interface shall implement those functions specified in the HLA Interface Specification for HLA compliant entities. The local interface to the entity/facility shall be at the user's local option. This interface to the Network shall be implemented in software using the Application Programmer's Interface (API) library provided from the Defense Modeling and Simulation Office with the current version of the Run-Time Infrastructure (RTI). Within the time management services specified in the HLA Interface Specification, only wall-clock time must be supported by the TSLA Network.

The requirements for interfacing which must occur so that the HLA Interface may connect to the existing entity or entity Facility Control are to be determined locally at each installation. The HLA interface shall be responsible for capturing the current values for each complex attribute data element specified in Appendix C for the given entity class to be interfaced. If the HLA Interface is being developed for a facility serviced by a Facility Control, this HLA Interface shall provide this function for all entities managed by the Facility Control.

The collection of data elements defined in Appendix C for each entity class may be considered a set of structured attributes. Appendix B includes a specification for a minimum update interval for each of these structured attributes. The HLA Interface shall publish the entire structured attribute for its entity class at an interval specified in Appendix B. The update interval shown in Appendix B is equal to the smallest interval for any of the data elements specified in Appendix C for that entity class. This update process shall be executed periodically, regardless of the change in state of any of the data elements in the structured attribute.

One allowable exception to this requirement is the *TTRadarPDUStruct* and the *AIRadarPDUStruct* data types. In this case, the missile guidance command data must be updated at 100 Hz. The remaining data (which is the bulk of the attribute) is only needed at 40 Hz. In this case, it is permissible to transmit the command data separate from the remaining radar data on the basis of required update rate.

As indicated in Appendix C, each publication of the structured attribute packet shall contain a time tag which shall indicate the time at which the data packet was assembled.

3.2.3.3. NETWORK INTERFACE

The HLA Interface shall connect to the commercial long-haul network using standard protocol methods for TCP/IP and Ethernet as listed in Section 2.2.2, or other protocols supported by the RTI.

3.2.3.4. DATA REDUCTION

At a minimum, each entity or Facility Control for an entity (ies) shall provide local test data reduction capabilities for any data that is generated or consumed by that entity. The raw data shall be stored locally as a backup and to serve as an offline diagnostic tool. Data reduction shall include the following functions:

1. Reference sensor entities shall report the measured position of entities in earth-centered coordinates according to the WGS84 standard. Any coordinate transformation from the native sensor coordinate system to the earth-centered system shall be performed by the Data Reduction Segment.
2. In some instances, it is necessary to compensate for the latency in transmission of data. In these cases, a filter-predictor shall be used to operate on the time-tagged data provided. This filter-predictor shall provide extrapolations of the data based on the state variable estimates produced by the filter. Specifications for the latency compensation method are provided in Appendix D.
3. All tracking sensors shall compute their own tracking error by comparing the target location reported by that sensor with that reported by the reference sensor entity.
4. In those instances where the weapon entity is implemented in a simulation, miss distance shall be computed by comparing the trajectory of the simulated weapon to the position of the target entity as reported by the reference sensor. The vector point of closest approach in 3-dimensional space shall be reported as the miss distance. The miss distance shall be reported in meters.
5. All sensor data shall be reported in the units and with the resolution specified in Appendix C. Any conversion of raw sensor data needed to convert to these standard units shall be performed by the Data Reduction Segment.
6. The Data Reduction Segment shall support a graphical user interface for local display of any data being reduced from the local data. This interface shall also provide local diagnostic capabilities to determine the readiness of the local entity and the local connection to the network.
7. The Date Reduction Segment shall support the Test Director data reduction requirements defined in Section 3.7.9, et.seq.

3.2.3.5. NETWORK MONITOR

A monitor shall be installed between the facility gateway and the bridge to the T1 multiplexer. This monitor shall detect and record 1) the number of packets passing in and out of the facility as a function of time, and 2) the number of packet fragments which are not completed.

3.2.4. EXTERNAL INTERFACE REQUIREMENTS

3.2.4.1. EXTERNAL INTERFACE: HUMAN OPERATOR INTERFACE

See Section 3.3.7 for general requirements for the human operator interface to the CDRSS.

3.2.5. PHYSICAL CHARACTERISTICS

N/A

3.2.5.1. PROTECTIVE COATINGS

No protective coatings are required for any of the segments of the TSLA network. All components will be operated in normal laboratory climate conditions.

3.2.6. SYSTEM QUALITY FACTORS

3.2.6.1. RELIABILITY

The TSLA Network Interface shall not degrade the reliability performance of the attached entities/facilities.

3.2.6.2. MAINTAINABILITY

The TSLA Network Interface shall not degrade the maintainability performance of the attached entities/facilities

3.2.6.3. AVAILABILITY

The TSLA Network Interface shall comply with the Roll Call requirements specified in section 3.2.1.1. The TSLA Network Interface shall not degrade the availability performance of the attached entities/facilities.

3.2.6.4. ADDITIONAL QUALITY FACTORS

3.2.7. ENVIRONMENTAL CONDITIONS

N/A

3.2.8. TRANSPORTABILITY

No special handling or transport requirements.

3.2.9. FLEXIBILITY AND EXPANSION

3.2.10. PORTABILITY

Not applicable

3.3. DESIGN AND CONSTRUCTION

3.3.1. MATERIALS

The TSLA Network shall use commercial parts and processes wherever possible.

3.3.1.1. TOXIC PRODUCTS AND FORMULATIONS

N/A

3.3.2. ELECTROMAGNETIC RADIATION

The segments developed to support the implementation of the TSLA Network shall be electromagnetically compatible with the pre-existing segments, the commercial communication network, and the RTI.

3.3.3. NAMEPLATES AND MARKINGS

Not applicable

3.3.4. WORKMANSHIP

Not applicable

3.3.5. INTERCHANGEABILITY

Any HLA compliant entity shall be replaceable by an equivalent HLA compliant entity.

3.3.6. SAFETY

3.3.7. HUMAN ENGINEERING

Display systems designed for operators and test observers will facilitate the observation of both a global view, in addition to specific views, of the test currently running or of previous tests. The human engineering program will ensure that the design of display systems support the ready interpretation of global and specific views. Software-driven user interfaces will be provided, where prompts, menus, and other aids will facilitate the operation and maintenance of system equipment. User interfaces will be intuitive, standard, uncomplicated, responsive, and flexible. In this manner, the complexity of interface operation will be reduced; the amount of training required for learning interfaces will be reduced; and a means of tailoring interfaces to meet changing mission requirements will be established. Consistency in GUIs and all other human-machine interfaces (with respect to comparable operator actions and commands) will be established. Data entry will be accomplished via a combination of nested menus, menu bars, icons, prompts, or direct input as consistent with the action(s) necessary for a task and the integrated overall design approach for user interfaces. Specific error messages corresponding to unacceptable and inappropriate operator inputs will be provided. Warning messages will be provided prior to execution of any operator commands that could adversely affect the system or system performance.

The control input shall be designed to minimize steps required for executing any command required during the real-time execution of the test. A graduated interface design shall be implemented. The graduation in capability shall permit simplified operation for novice users, at the expense of efficiency. For more advanced users, the interface shall provide less obvious, but more efficient methods for control and display.

3.3.8. NUCLEAR CONTROL

Not applicable.

3.3.9. SYSTEM SECURITY

All transmissions of data from an entity or a facility controller shall be encrypted for information security. The level of security shall be the same throughout the entire federation and shall be the maximum level required by any entity participating in the federation.

3.3.10. GOVERNMENT FURNISHED PROPERTY USAGE

None

3.3.11. COMPUTER RESOURCE RESERVE CAPACITY

The reserve capacity of the computing resources utilized to implement the Network Interface and the Data Reduction segments shall be at least 50%.

3.4. DOCUMENTATION

Documentation for the TSLA Network shall consist of the following:

a) Operator’s Manual

b) Hardware Reference Manual

c) Software Reference Manual

3.5. LOGISTICS

Not Applicable.

3.6. PERSONNEL AND TRAINING

3.6.1. PERSONNEL

In addition to existing facility personnel, the following position types shall must be supported in order to operate the TSLA Network.

a) Test Control Monitor

b) Test Data Analyst

c) Test Performance Analyst

3.6.2. TRAINING

The TSLA Network developer shall be responsible for training. Training shall include the curriculum, devices, and materials necessary to develop skills in the following areas:

1. How to install and configure the TSLA Network Interface
2. How to operate and maintain the TSLA Network Interface hardware and software
3. How to develop scenario simulations
4. How to develop, conduct, and analyze the results of a test

The TSLA Network developer shall provide the training devices, materials, aids, and schedules as defined in the SOW and pertinent CDRLs.

3.7. CHARACTERISTICS OF SUBORDINATE ELEMENTS

3.7.1. SEGMENT 1: RADAR ENTITY

3.7.1.1. ENCAPSULATION REQUIREMENTS

Req. 5. The Radar entity shall be encapsulated as shown in Figure 7, unless the radar is encapsulated within a facility, in which case the facility shall provide the injection and monitoring services necessary for the entity to participate in the data domain.

Req. 6. A Data Mode Monitor shall be provided to produce reports of the control settings made within the radar asset.

Req. 7. An Radar RF Monitor shall be provided to produce reports of the measured RF mode of the radar asset.

Req. 8. A Target Signal Injection Unit shall be provided to produce RF input to the radar which simulates the returns of the platform being observed by the radar.

Req. 9. An ECM Injection Unit shall be provided to produce RF input to the radar which simulates the jamming environment present at the radar.

Req. 10. A Clutter Injection Unit shall be provided to produce RF input to the radar which simulates the clutter environment present at the radar.

Req. 11. A Chaff Injection Unit shall be provided to produce RF input to the radar which simulates the chaff environment present at the radar.

Req. 12. An RF Environment Monitor shall be provided to confirm the correctness of the RF input to the radar produced by the various injection units.

Req. 13. The Radar entity shall be the owner of the radar antenna pattern data.

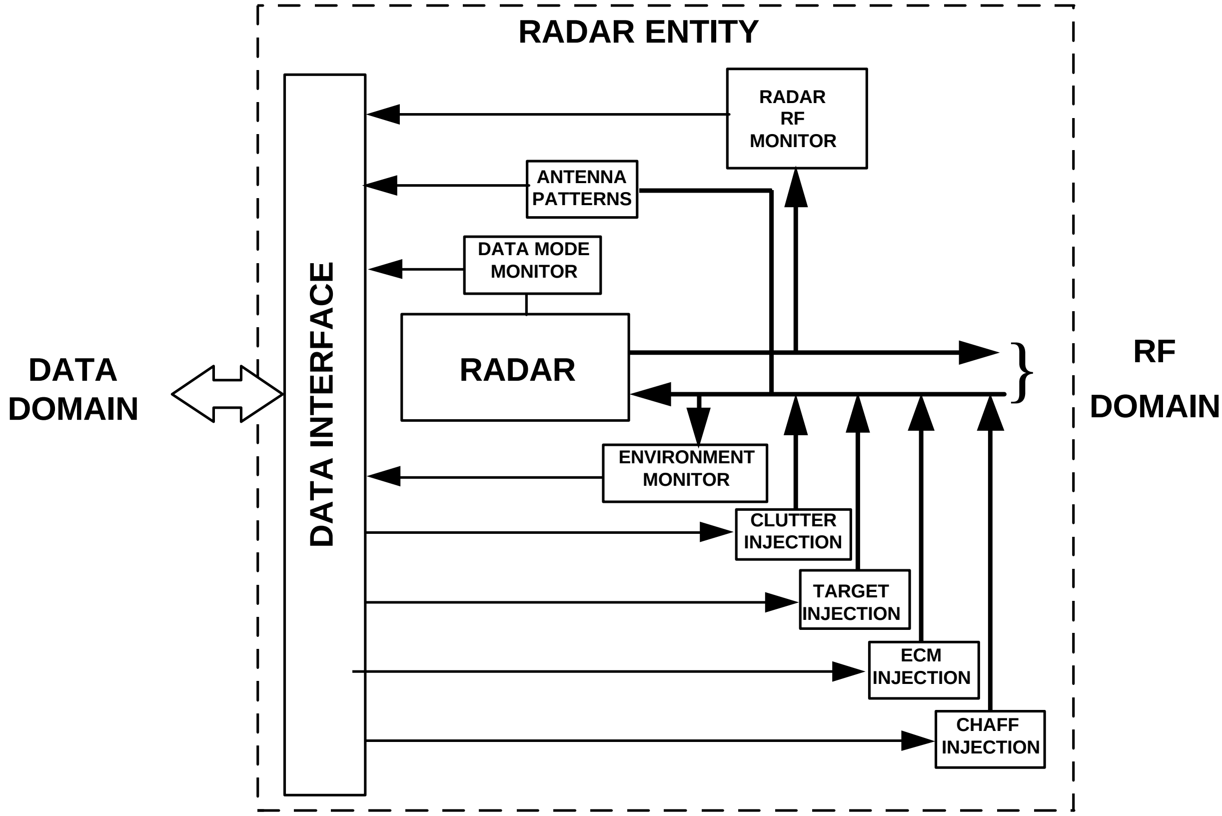

Figure 7. Encapsulation of Radar for Insertion in Data Domain.

3.7.1.2. INSTRUMENTATION REQUIREMENTS

The radar entity shall serve as the custodian for the measured data on its antenna patterns.

Prior to a test, the radar's measured antenna patterns shall be exported by the Test Director to those entities which may require the patterns for calculations (e.g., the ECM entity). The antenna pattern block in Figure 7 is used to point out the source or logical owner of the data.

A radar mode monitor shall be provided to determine and report the current operating mode of the radar. Depending on the radar entity, this monitor may observe the physical position of mechanical switches used to control the mode of the radar, or it may monitor data which serves to control the operating mode. In either case, this monitor shall report to the HLA Network Interface, a unique identifier code for the detected operating mode.

A radar RF monitor shall be provided to perform an RF verification of the radar operating mode. This instrumentation is required only for open-air range assets when RF interaction is a part of the test, (i.e., where it is important to make certain that the signal environment contributed by this asset is appropriate for the other assets involved in the test). This monitor shall measure and record those waveform parameters needed to make the determination of correctness of the mode reported by the radar data mode monitor. Since it may take some time to collect the raw RF data to determine the operating mode, this monitor in not considered to run in true real time, but rather as a confirmation for the radar data mode monitor which operates on the switch settings.

Signal injection unit(s) shall be provided to inject synthetic target returns, ECM, clutter and chaff. It is permissible to combine these synthesizer functions into a composite signal generator. Each piece of injection equipment shall be controllable by data obtained from other entities. For example, the Target Injection Unit shall obtain radar-range-equation parameters from the appropriate platforms.

An Environment Monitor shall be provided to confirm the correctness of each of the signal injection source inputs to the radar. This monitor shall operate using the data which controls the signal injection units. Since it may take some time to collect the raw RF data to determine the operating mode of the signal injection units, this monitor in not considered to run in true real time, but rather as a confirmation for the operating mode reported by the injection units.

3.7.1.3. PUBLICATION REQUIREMENTS

Radar entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Radar class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Radar entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.1.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each radar entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B. Note that for TTR Radar type and AI Radar type, guidance command updates are required at a higher rate than the remaining attributes.

3.7.2. SEGMENT 2: THREAT ENTITY

3.7.2.1. ENCAPSULATION REQUIREMENTS

Threat types are delineated based on the particular guidance method. A Threat entity in the TSLA Network may take one of three forms:

1. live fire on a test range,
2. a virtual simulation in a chamber/hot-bench, or
3. a constructive simulation.

In the case of live fire, the components of the missile clearly cannot be distributed. For this case, the missile operates entirely in the real environment. Of course, position reference monitors and RF environment monitors may be needed so that the missile can be represented to any distributed entities. When the missile is operated on a bench or a flight table, the seeker head, the autopilot and a flight table (if used) will most likely be collocated. If such tight linking requirements exist, they can be handled using facility-based communication under a Facility Controller. Finally, in the constructive case, components of the missile could be geographically separated. The seeker and airframe are two separate TSLA entities and could be encapsulated in a single facility if tight linking is required.

For artillery, the TTR/AIR system calculates the pointing angle or pointing lead angle required for the selected target. Artillery entities, like the missile entities are constructed from two separate entities.

Threats are assumed to be closely associated with a TTR or AIR system. In the case of an active missile, the TTR/AIR system is usually deployed within the airframe of the missile. From a data perspective, however, the TTR/AIR system does not need to be contained within the same facility as the missile airframe. With the exception of beacon properties on a command guided missile, the publication requirements of all the threat entities are similar and shall be encapsulated as shown in Figure 8.

3.7.2.2. INSTRUMENTATION REQUIREMENTS

The composite of all required instrumentation for the threat entity is shown in Figure 8. Reference sensor support shall be provided for live missile players to report missile position to the other players participating in the data domain. All types of command guided, and some semiactive and active missiles require a command input. For these, a command interface shall be provided to translate the data received over the network into the signal format required in the missile command receiver. A beacon return generator shall be provided for all missiles which are, at some point in their flyout, tracked by the tracking radar. In certain systems, this beacon response is simply a pulse reply to the missile track radar signal. For these systems, the beacon data output is a signal level reply, equivalent to the RCS used in the detection range calculation of passive targets. In other systems, the output from the beacon transponder is a data stream, providing various types of status to the tracking radar.

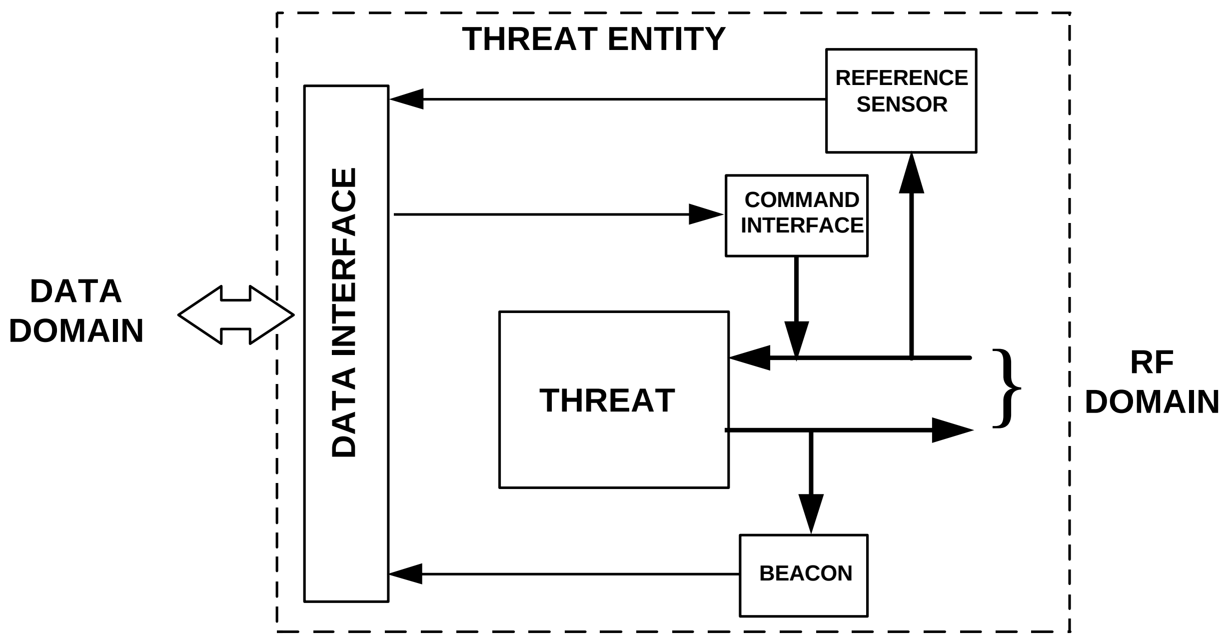

Figure 8. Encapsulation of a Threat Entity.

3.7.2.3. PUBLICATION REQUIREMENTS

Threat entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Threat class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Threat entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.2.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Threat entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.3. SEGMENT 3: ECM ENTITY

3.7.3.1. ENCAPSULATION REQUIREMENTS

An ECM system interacts with the environment through the RF domain. Conversion to the data domain will be accomplished using the encapsulation illustrated in Figure 9.

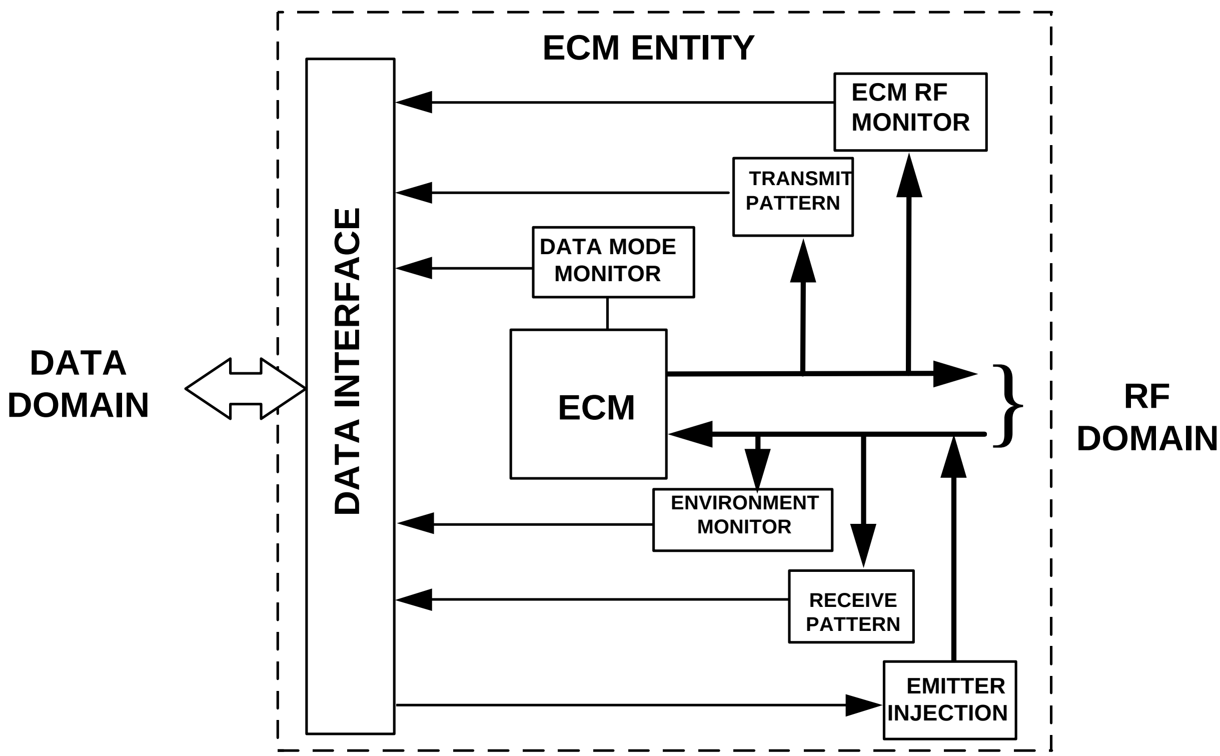

Figure 9. Encapsulation for the ECM Entity.

3.7.3.2. INSTRUMENTATION REQUIREMENTS

The ECM entity shall serve as the custodian for the measured data on its antenna patterns.

Prior to a test (i.e., prior to federation execution), the ECM measured antenna patterns shall be exported by the Test Director to those entities which may require the patterns for calculations (e.g., the Radar entity). The antenna pattern block in Figure 9 is used only to point out the source or logical owner of the data.

An ECM mode monitor shall be provided to determine and report the current operating mode of the ECM. Depending on the ECM entity, this monitor may observe the physical position of mechanical switches used to control the mode of the ECM, or it may monitor data which serves to control the operating mode. In either case, this monitor shall report to the HLA Network Interface, a unique identifier code for the detected operating mode.

An ECM RF monitor shall be provided to perform an RF verification of the ECM operating mode. This instrumentation is required only for open-air range assets when RF interaction is a part of the test, (i.e., where it is important to make certain that the signal environment contributed by this asset is appropriate for the other assets involved in the test). This monitor shall measure and record those waveform parameters needed to make the determination of correctness of the mode reported by the ECM data mode monitor. Since it may take some time to collect the raw RF data needed to determine the operating mode, this monitor is not considered to run in true real time, but rather as a confirmation for the ECM data mode monitor which operates on the switch settings or control data.

Emitter Injection Unit(s) (EIU) shall be provided to inject synthetic emitter inputs. The emitters to be simulated by the EIU shall include a) those emissions of other specific entities in the federation, and b) the collection of background emitters in the scenario of interest. The EIU output for the specific federation emitters shall be controllable by data obtained from Radar entities participating in the federation. The background emitter outputs shall be controlled by a local script loaded into the SIU.

An Environment Monitor shall be provided to confirm the correctness of each of the emitter inputs to the ECM. This monitor shall operate using the data which controls the Emitter Injection units. Since it may take some time to collect the raw RF data to determine the operating mode of the Emitter Injection units, this monitor in not considered to run in true real time, but rather as a confirmation for the operating mode reported by the injection units.

3.7.3.3. PUBLICATION REQUIREMENTS

ECM entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the ECM class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

ECM entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.3.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each ECM entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.4. SEGMENT 4: PLATFORM ENTITY

3.7.4.1. ENCAPSULATION REQUIREMENTS

A live platform entity must be encapsulated in order to live in the data domain of the a TSLA Network. The platform is considered to possess the following attributes: position, orientation, and radar cross section. Each of these attributes must be provided to the data domain. The method for encapsulating the platform entity shall be as shown in Figure 10.

The required RCS data will be measured and transferred to the Affected facilities prior to the federation execution.

3.7.4.2. INSTRUMENTATION REQUIREMENTS

The Platform entity shall serve as the custodian for the measured data on its radar cross section.

Prior to a test (i.e., prior to federation execution), the measured RCS data shall be exported by the Test Director to those entities which may require the data for calculations (e.g., the Radar entity). The RCS block in Figure 9 is used only to point out the source or logical owner of the data.

For live Platform entities, sensors shall be provided to determine the location and orientation of the entity. These sensors could be some combination of reference radars, GPS receivers, inertial navigation systems, and platform attitude indicators. The sensors shall provide the data needed to encapsulate the platform into the data domain.

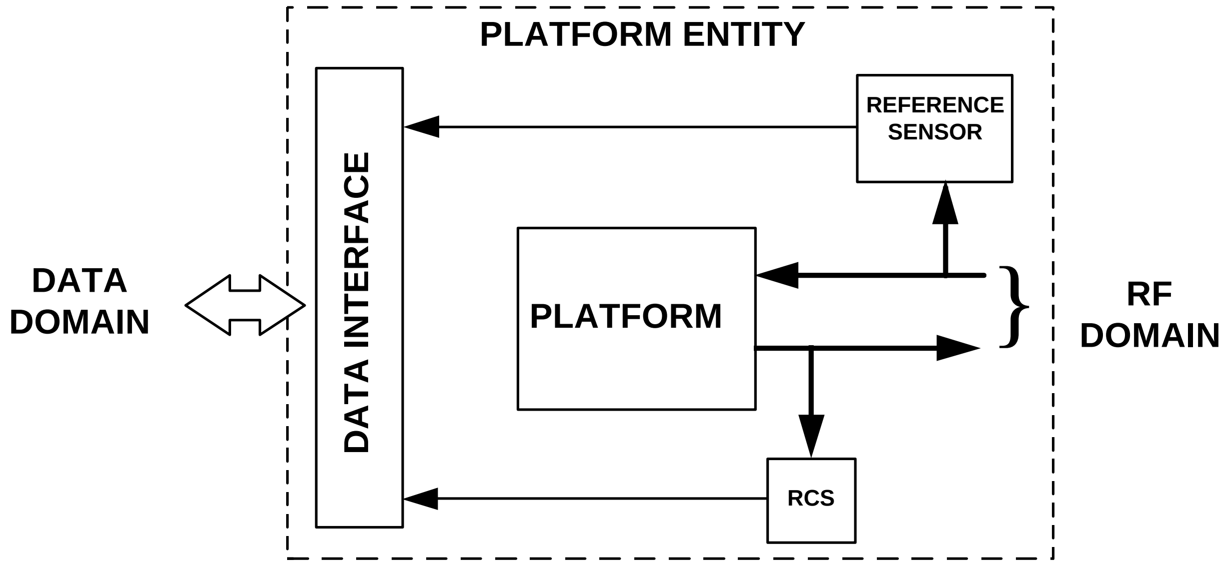

Figure 10. Encapsulation of the Platform Entity

3.7.4.3. PUBLICATION REQUIREMENTS

Platform entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Platform class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Platform entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.4.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Platform entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.5. SEGMENT 5: C^3 ENTITY

3.7.5.1. ENCAPSULATION REQUIREMENTS

The C^3 system shall be encapsulated for use in the TSLA Network environment as shown in Figure 11. For those cases where certain assets are tightly coupled within a facility (e.g., C^3 and a TAR) and require a direct, in-the-facility data path that is either too specialized in terms of protocol or too high in bandwidth to be realistically supported by network communications, that set of entities shall be considered to reside within a facility. In this case, facility encapsulation like that shown in Figure 12 shall be used. The actual encapsulation of each entity (e.g., the data) and the overall encapsulation of the facility (e.g., all potential communication paths) are issues that must be resolved during federation development.

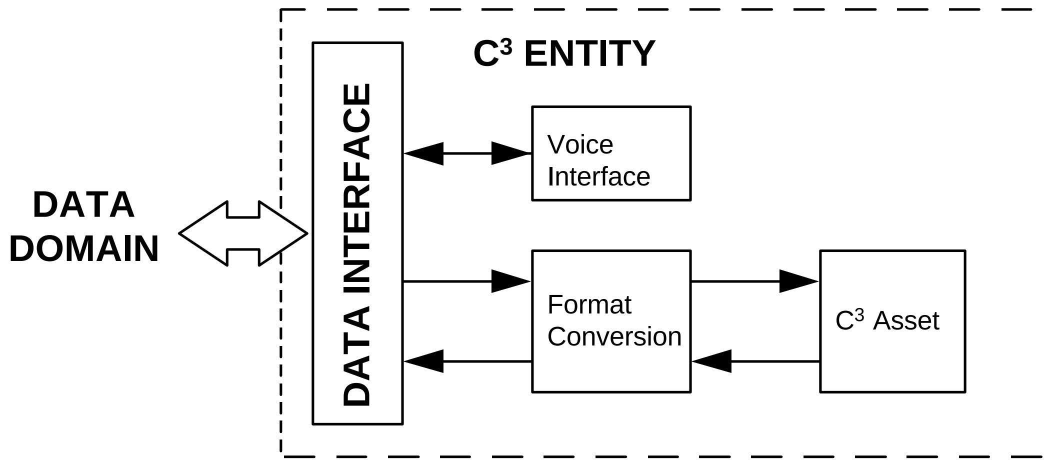

Figure 11. Encapsulation of the C^3 Entity

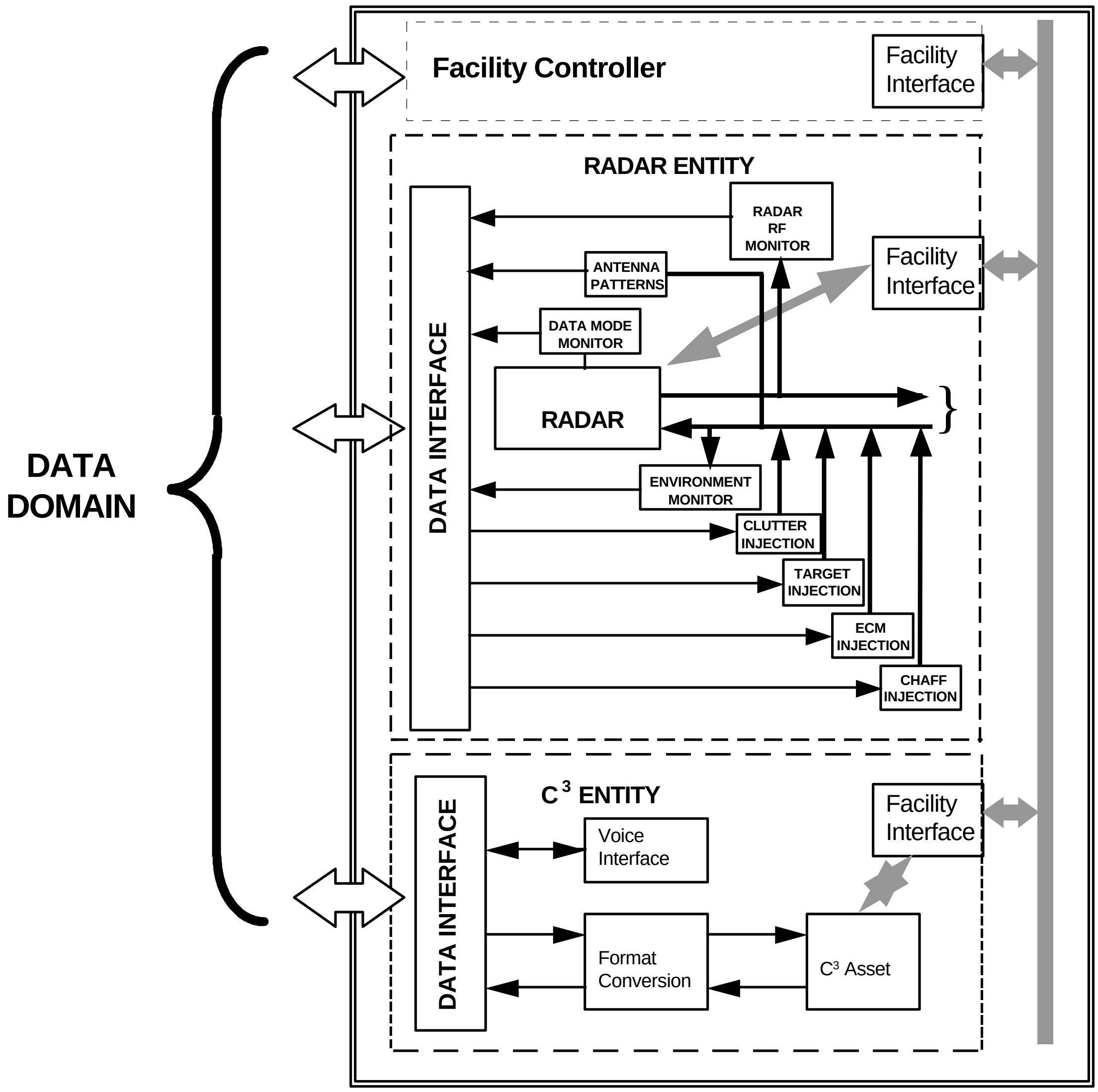

Figure 12. Encapsulation example for a facility with C^3 and TAR

3.7.5.2. INSTRUMENTATION REQUIREMENTS

3.7.5.2.1. DATA INTERFACE REQUIREMENTS

An interface shall be provided to convert between signal formats of the tactical C^3 system and the network. The exact requirements will depend on the particular implementation of the tactical C^3 system. The interface must use the serial data stream on the tactical C^3 side of the interface and network data structures on the network side for both incoming and outgoing messages.

The native interfaces typically include initial synchronization strings which allow the receiving end to synchronize to the rate of the incoming data. In addition to a synchronization pattern, the messages usually include an initial character string which identifies the beginning of the message. In general, the messages will also include data bits for error detection, error correction, and encryption.

These asset-specific requirements shall be designed into the data interface so that it will not be necessary to transmit the additional data over the network.

3.7.5.2.2. VOICE COMMUNICATIONS INTERFACE

The tactical C^3 entity shall provide a voice communications interface to provide threat-faithful voice communications between the various players in the threat air defense system. The voice links are an integral part of the mission of the C^3 system but need not be carried over the TSLA Network.

3.7.5.3. PUBLICATION REQUIREMENTS

C^3 entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the C^3 class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

C^3 entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.5.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each C^3 entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.6. SEGMENT 6: STANDALONE ENVIRONMENT MONITOR ENTITY

3.7.6.1. ENCAPSULATION REQUIREMENTS

The Stand-Alone Environment Monitor is intended to function as a monitor of the real environment experienced by the live entities in the TSLA federation. In a sense, it *is* the encapsulating device which moves the extraneous RF environment into the data domain. The Stand-Alone Environment Monitor does not require encapsulation.

3.7.6.2. INSTRUMENTATION REQUIREMENTS

Not Applicable. The Stand-Alone Environment Monitor *is* the instrumentation.

3.7.6.3. PUBLICATION REQUIREMENTS

Stand-Alone Environment Monitor entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Stand-Alone Environment Monitor class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Stand-Alone Environment Monitor entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.6.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Standalone Environment Monitor entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.7. SEGMENT 7: COMMUNICATION ECM ENTITY

3.7.7.1. ENTITY ENCAPSULATION REQUIREMENTS

Only virtual or constructive implementations shall be permitted on the TSLA Network. These implementations are naturally confined to the data domain so that the encapsulation will be a standard part of the entity. A block diagram indicating the required deployment method for this entity is shown in Figure 13. The dotted arrow within the entity is meant to denote the normal path for information flow in the absence of a Communications ECM Entity. For the purposes of the TSLA Network it is expected that the communications ECM function will be contained within the entity itself so that no other inputs are required.

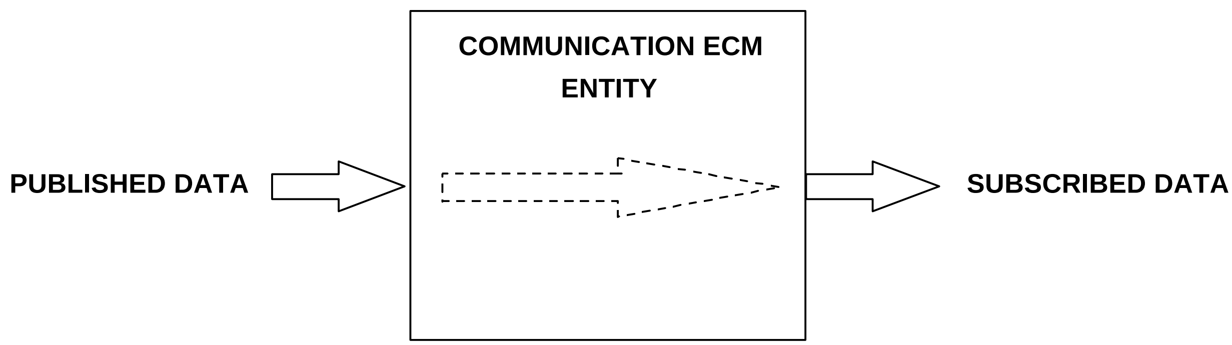

Figure 13. Communications ECM Entity

3.7.7.2. INSTRUMENTATION REQUIREMENTS

Not Applicable

3.7.7.3. PUBLICATION REQUIREMENTS

Communication ECM entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Communication ECM class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Communication ECM entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.7.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Communication ECM entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.8. SEGMENT 8: FACILITY CONTROLLER ENTITY

3.7.8.1. ENCAPSULATION REQUIREMENTS

No encapsulation is required for the Facility Controller. It resides in the data domain in its natural state.

3.7.8.2. INSTRUMENTATION REQUIREMENTS

Not applicable.

3.7.8.3. PUBLICATION REQUIREMENTS

Facility Controller entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Facility Controller class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Facility Controller entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.8.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Facility Controller entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.9. SEGMENT 9: TEST DIRECTOR ENTITY

3.7.9.1. ENCAPSULATION REQUIREMENTS

Not applicable.

3.7.9.2. INSTRUMENTATION REQUIREMENTS

3.7.9.2.1. DATA COLLECTION REQUIREMENTS

The Test Director Facility shall have the capability to monitor all network traffic exchanged between entities during a test. The data are necessary to perform a quick-look assessment of the SUT, as well as the overall health of the network, during the test. Data logging requirements necessary to perform a detailed analysis of the test are determined during federation development, but, at a minimum, shall include those requirements listed in Section 3.1.1.9.

3.7.9.2.2. INSTRUMENTATION REQUIREMENTS

Req. 14. The Test Director Facility shall have the ability to remotely control all instrumentation systems.

The Test Director Facility does not have any specific instrumentation requirements. The Test Director Facility does however interact with all test instrumentation residing at other entities/facilities. The Test Director Facility shall have the ability to remotely control all instrumentation systems using measurement scripts. These scripts shall be downloaded prior to the start of a test.

3.7.9.2.3. REAL-TIME DATA COLLECTION, PROCESSING, ANALYSIS, AND DISPLAY

Req. 15. The Test Director Facility shall have the ability to display a scenario map while the test is proceeding.

Req. 16. The Test Director Facility shall have the ability to display the test status while the test is proceeding.

Req. 17. The Test Director Facility shall have the ability to display a SUT display while the test is proceeding.

Req. 18. The Test Director Facility shall have the ability to display SUT performance measures according the requirements arising from federation development.

Req. 19. The Test Director Facility shall have the ability to display a filter center plot board while the test is proceeding.

Req. 20. The Test Director Facility shall have the ability to display the network status while the test is proceeding.

Req. 21. The Test Director Facility shall have the capability to communicate directly to facility personnel using voice communications.

3.7.9.2.3.1. TEST VISUALIZATION

The Test Director Facility shall be capable of providing full test visualization to the Test Director. Test visualization shall be at a sufficient level of detail to provide the director the information necessary to determine that the test is proceeding as planned.

A scenario map similar to the map shown in Figure 14 shall be provided. This display shall provide at least the following information:

1. Geographical region including terrain, physical landmarks, latitude/longitude grid,
2. Threat positions and state (off, search/acquisition, track, missile launch),
3. SUT position and attitude vs. time, and
4. Airborne Interceptor (AI) position and attitude vs. time.

A status window shall be provided that contains the following information:

1. Time of Day, Scenario Start Time, Elapsed Time Since Scenario Start Time,
2. Network Status, and
3. Error Messages.

This information provides the Test Director with a status of the test and the health of the test setup. Any problems or errors shall be reported in this display. An example of the status display is shown in Figure 14.

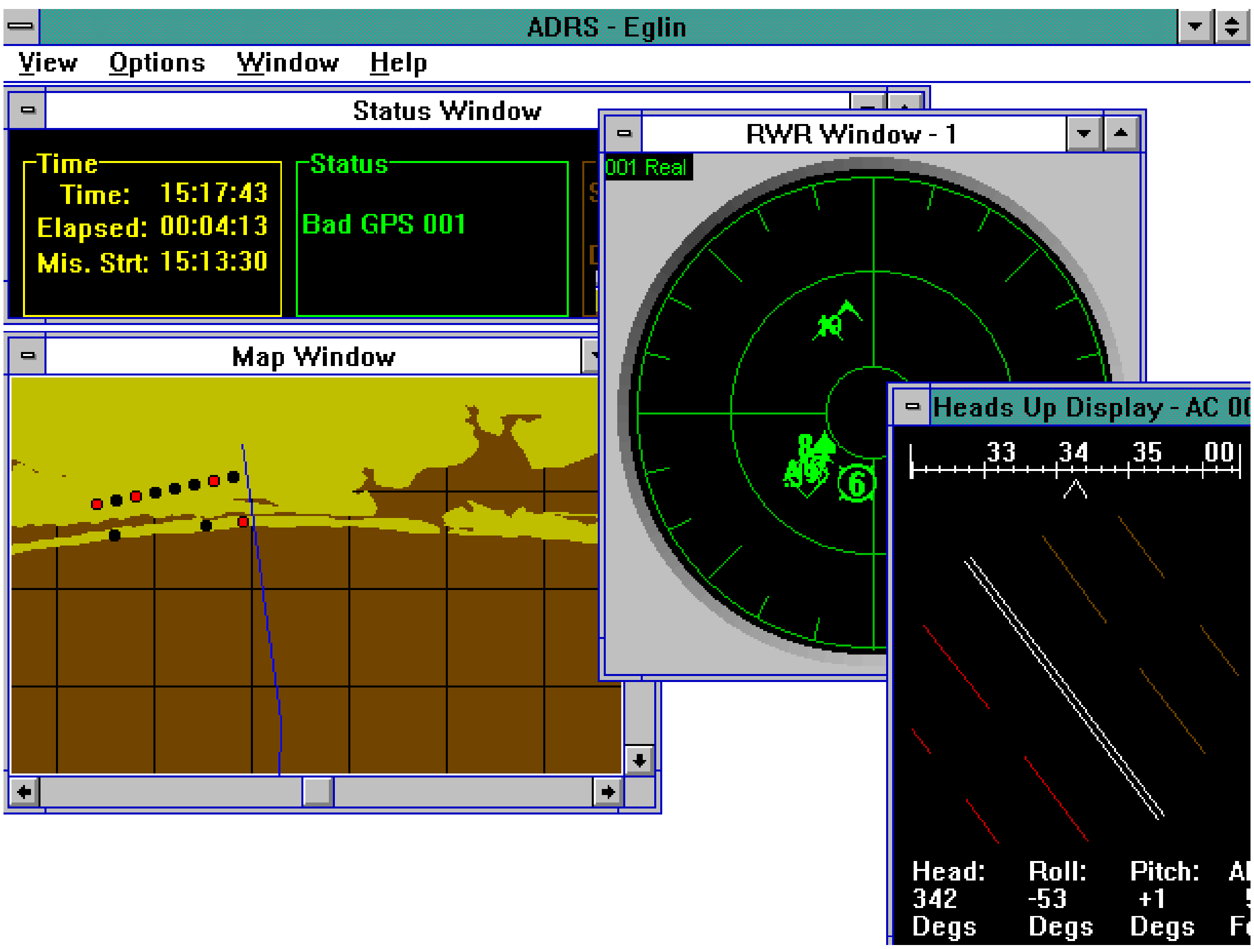

Figure 14. Example Scenario Map Status Display for Test Visualization.

A simulated SUT display containing both actual and expected responses shall be provided. The simulated displays shall be as close to the actual SUT display to determine if the SUT is responding as expected. This would include a time history of SUT responses in the form of color-coded histograms.

A filter center plot board display shall be provided similar to the plot board shown in Figure 15. The plot board is used to monitor the status of C^3 assets. This board shall contain a plot of all active target tracks and their status (e.g., tracking, assigned to terminal threat).

A display containing a tabular list of active emitters shall be provided. The displays list of emitters indicates which emitters are supposed to be active along with another indication as to which emitters have been verified as radiating in the correct mode. The ECM technique verification procedure will use a list of active techniques to perform its function. The emitter list shall also contain a field to indicate the ECM technique and the verification status of the technique.

A network status display shall be provided. This display shall indicate all active entities/facilities on the network and their general status.

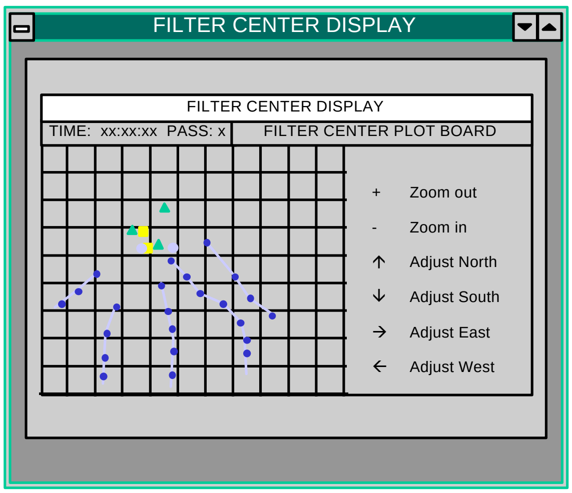

Figure 15. Filter Center Plot Board Example.

3.7.9.2.3.2. SUT Assessment

Depending on the design of the EW test, different measures of SUT performance are important. The Test Director Facility shall provide the capability to display the following SUT performance measures:

Response Time for Signal Intercept,
Threat Detection/ Deletion Range,
Correct Emitter ID,
Response Time for Correct Final ID,
Correct Threat Prioritization,
DF Accuracy,
ECM Technique Response Time

Correct Technique Selection,
ECM Technique Generation,
ECM Technique Activation/Deactivation,
Multi-Threat Jamming Response,
Correct Technique Parameter Emission,
Jamming-to-Signal Ratio,
Engagement Time/Envelope Reduction,
Tracking Error,
Breaklock/ Track Loss,
Reduction in Missile Launches/ Projectiles Fired,
Missile/ Projectile Miss Distance,
Reduction in Lethality (RIL),
Probability of Kill, and
Net Reduction in Lethality.

The ability of the Test Director Facility to display these measures does not imply the requirement to compute these measures.

3.7.9.2.4. USER INTERFACE REQUIREMENTS

The Test Director Facility shall provide real-time controls and displays for the test. The Test Director will be required to quickly interpret displayed data, make decisions, and execute commands. A reconfigurable, graphical user interface (GUI) shall be implemented for this interface to support rapid decision making. Reconfigurablity shall include specifying the number of windows, the size of each window, the location of the window, and the type of data displayed. The ability to select axis scaling for all displays shall be provided. Axis scaling includes changing the scenario map region (e.g., zoom in, zoom out, move display), changing the time resolution on displays, and changing the axis resolution on the performance measure displays.

3.7.9.2.5. VOICE COMMUNICATION REQUIREMENTS

Voice link capability to other entity/facility personnel shall be provided for test coordination. The purpose for this voice link is not part of the federation execution, but rather a part of the structure for federation development.

3.7.9.3. PUBLICATION REQUIREMENTS

Test Director Facility entities shall be capable of publishing the attributes specified in Appendix C. Each entity in the Test Director Facility class is specified individually in Appendix C. A given entity type shall only be required to publish the attributes defined for that type.

Test Director Facility entities shall publish attributes in the format of Appendix C and with the frequency specified in Appendix B. A given entity type shall only be required to publish the attributes defined for that type.

3.7.9.4. DATA SIZE AND UPDATE REQUIREMENTS

The data packet for the structured attribute for this entity, exclusive of the transport overhead, required for each Test Director Facility entity type shall not exceed the value indicated in the *bytes* column for that entity in Appendix B. The update interval for this attribute shall not exceed the value indicated in the *Update Condition* column in Appendix B.

3.7.10. SEGMENT 10: RUN TIME INFRASTRUCTURE

The Run Time Infrastructure is a distributed operating system which provides connection and data services to all of the known subscribers on the TSLA network. The RTI shall conform to the specifications listed below in providing services. All TSLA entities shall conform to these specifications for interfacing and service requests.

HLA Interface Specification
HLA Object Model Template
HLA Rules

3.7.11. SEGMENT 11: COMMUNICATION INFRASTRUCTURE

The communication transport method for RTI is expected to be one of more of the packet-switched protocols (e.g., TCP/IP, UDP, ATM, etc.) At a minimum, a TSLA entity shall provide the capability of interfacing to the communication network using TCP/IP protocol in accordance with the standards in Section 2.2.2.

3.8. PRECEDENCE (OF REQUIREMENTS)

TBD

3.9. QUALIFICATION

This section defines the set of qualification methods for verification to ensure that the requirements specified in Section 3 have been met. In general, the following qualification methods will be used to verify each of the requirements.

1. Demonstration (D): The operation of the systems or part of the system, that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
2. Test (T): The operation of the system, or a part of the system, using instrumentation or other special test equipment to collect data for later analysis.
3. Analysis (A): The processing of accumulated data obtained from qualification methods. Examples are reduction, interpolation, or extrapolation of test results.
4. Inspection (I): The visual examination of system components, documentation, etc.
5. Special Qualification Methods (S): any special qualification methods for the system, such as special tools, techniques, procedures, facilities, acceptance limits, use of standard samples, preproduction or periodic production samples, pilot models, or pilot lots.

Table 2 lists the qualification methods for the TSLA Network requirements contained in Section 3 of this specification.

Table 2. Qualification Methods.

Section Number	Requirement	Qualification Method
3.2.1	Performance Characteristics	I, D, T, A
3.2.3.1	Facility Control	I, D, T, A
3.2.3.2	HLA Network Interface	I, D, T, A
3.2.3.3	Network Interface	I, D, T, A
3.2.3.4	Data Reduction	I, D, T, A
3.2.6	System Quality Factors	T
3.4	Documentation	D
3.7.9	Segment 9: Test Director Entity	I, D, T, A
3.6.2	Training	D

3.10. STANDARD SAMPLE

None Required

3.11. PREPRODUCTION SAMPLE

None Required

4. QUALITY ASSURANCE PROVISIONS

4.1. RESPONSIBILITY FOR INSPECTION

The Joint Test Force of the JADS Program Office will have overall responsibility for tests and inspection of any items developed in response to this specification.

4.2. SPECIAL TESTS AND EXAMINATIONS

An Acceptance Test Procedure shall be required for each item developed in response to this specification. Successful Acceptance Tests shall be completed prior to acceptance of any items developed in response to this specification, and prior to their use in any EW T&E federations.

4.3. REQUIREMENTS CROSS REFERENCE

Acceptance Test Procedures shall be prepared for each item developed in response to this specification. Those procedures shall insure that the item is in compliance with all relevant requirements of Sections 3 and 5 of this specification.

5. PREPARATION FOR DELIVERY

5.1. HARDWARE

All hardware shall be packaged and/or crated for delivery in accordance with appropriate commercial standards by the vendor.

5.2. SOFTWARE

As a minimum, all magnetic media containing source and/or object code for the TSLA network interface and all program code listings shall be labeled as follows:

1. Program/System Name
2. Program identification number (CSCI Number)
3. Version Identifier
4. Version Date

The version scheme described above shall be in accordance with the TSLA system program configuration and data management plans as applicable when the software is delivered.

6. NOTES

6.1. GLOSSARY

AAA	Anti-Aircraft Artillery
AIR	Airborne Interceptor Radar
BIT	Built-In Test
C^3	Command, Control, and Communication
C4I	Command, Control, Communications, Computers, and Intelligence
CD-R	Compact Disk Recordable
COTS	Commercial Off-the-Shelf
DIS	Standard for Distributed Interactive Simulation
DMSO	Defense Modeling and Simulation Organization
DOD	Department of Defense
ECM	Electronic Countermeasure
EW	Electronic Warfare
EWR	Early Warning Radar
FOM	Federation Object Model
FTP	File Transfer Protocol
GCI	Ground Control Intercept Radar
GPS	Global Positioning System
GUI	Graphical User Interface
HFR	Height Finder Radar

HITL	Hardware (Human) In The Loop
HLA	High-Level Architecture
IADS	Integrated Air Defense System
IEEE	Institute for Electronic and Electrical Engineers
JADS	Joint Advanced Distributed Simulation
MWR	Missile Warning Receiver
MWX	Missile Warning Radar
OMT	Object Model Template
PDU	Protocol Data Unit
QA	Quality Assurance
RCS	Radar Cross Section
RF	Radio Frequency
RID	RTI Initialization Data
RTI	Run-Time Infrastructure
RWR	Radar Warning Receiver
SARH	Semi-Active Radar Homing
SOM	Simulation Object Model
SUT	System Under Test
T&E	Test and Evaluation
TAR	Target Acquisition Radar
TDM	Time-Division Multiplexed
TDMA	Time-Division Multiple Access
TEL	Transporter Erector Launcher
TELAR	Transporter Erector Launcher and Radar
TES	Target Engagement System
TSLA	Threat Simulator Linking Activity
TSPI	Time Space Position Information
TTR	Target Tracking Radar
WGS	World Geographic Society

APPENDIX A. TSLA OBJECT CLASS STRUCTURE TABLE

Object Class Structure Table		
Entity	Radar Base	EW Radar
		HF Radar
		TA Radar
		TT Radar
		AI Radar
		MW Radar
	Threat Base	Command Guided Missile
		Active Missile
		Artillery
		SemiActive Missile
	Warning Base	Radar Warning Receiver
	ECM Base	Jammer
		Towed Jammer
		Chaff Dispenser
	C3	
	Stand Alone Monitor	
	Comm ECM	
	Platform	
Facility Controller		
Test Director		

APPENDIX B. TSLA FOM ATTRIBUTE PARAMETER TABLE

Attribute/Parameter Table										
Object/Interaction	**Attribute/Parameter**	**Datatype**	**Card**	**Units**	**Res**	**Accuracy**	**Accuracy Condition**	**Update Type**	**Update Condition**	**Bytes**
Entity.Radar_Base.EW_Radar	EWRadarPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	53
Entity.Radar_Base.HF_Radar	HFRadarPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	53
Entity.Radar_Base.TA_Radar	TARadarPDUStruct	Complex Data Type						conditional	0.05 sec update interval passed	53
Entity.Radar_Base.TT_Radar	TTRadarPDUStruct	Complex Data Type						conditional	0.025 sec update interval passed 0.01 sec for Guidance	151
Entity.Radar_Base.AI_Radar	AIRadarPDUStruct	Complex Data Type						conditional	0.025 sec update interval passed 0.01 sec for Guidance	135
Entity.Radar_Base.MW_Radar	MWRadarPDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	37
Entity.ThreatBase.CommandGuidedMissile	CommandGuidedMissilePDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	71
Entity.ThreatBase.ActiveMissile	ActiveMissilePDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	71
Entity.ThreatBase.Artillery	ArtilleryPDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	71
Entity.ThreatBase.SemiActiveMissile	SemiActiveMissilePDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	71
TestDirector	TestDirectorPDUStruct	Complex Data Type						conditional	1.0 sec update interval passed	46
FacilityController	FacilityControllerPDUStruct	Complex Data Type						conditional	1.0 sec update interval passed	46
Entity.WarningBase.RadarWarningReceiver	RadarWarningReceiverPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	17
Entity.ECMBase.ChaffDispenser	ChaffDispenserPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	90
Entity.ECMBase.Jammer	JammerPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	93
Entity.ECMBase.TowedJammer	TowedJammerPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	105
Entity.C3	C3PDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	86
Entity.StandAloneEnvironmentMonitor	StandAloneEnvironmentMonitorPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	15
Entity.CommunicationECM	CommunicationECMPDUStruct	Complex Data Type						conditional	0.1 sec update interval passed	89
Entity.Platform	PlatformPDUStruct	Complex Data Type						conditional	0.025 sec update interval passed	104

APPENDIX C. TSLA COMPLEX DATATYPE TABLE

See TSLA Network Requirements Specification 28 February 1997, Appendix A for a description of the information in each of these complex data types.

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
EWRadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	scan_period	long		ms	1		
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	begin_pulse	bool					
	checksum	unsigned long					
HFRadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_command	Complex Type					
	nod_period	long		ms	1		
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	begin_pulse	bool					
	checksum	unsigned long					
TARadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	boresight_azimuth	int		deg	0.1		
	boresight_elevation	int		deg	0.1		
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	checksum	unsigned long					
TTRadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	boresight_azimuth	int		deg	0.1		
	boresight_elevation	int		deg	0.1		
	track_error_azimuth	int		deg	0.1		
	track_error_elevation	int		deg	0.1		
	track_error_range	long		m	0.1		
	jammer_to_signal_ratio	int		db	0.1		
	signal_to_clutter_ratio	int		db	0.1		
	op_command	Complex Type					
	op_response	Complex Type					
	engagement_result	enum	EngagementResultEnum				
	miss_distance_x	long		m	0.1		
	miss_distance_y	long		m	0.1		
	miss_distance_z	long		m	0.1		
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	track_status	bool					
	checksum	unsigned long					
AIRadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	platform_serial_number	int					
	boresight_azimuth	int		deg	0.1		
	boresight_elevation	int		deg	0.1		
	track_error_azimuth	int		deg	0.1		
	track_error_elevation	int		deg	0.1		
	track_error_range	long		m	0.1		
	jammer_to_signal_ratio	int		db	0.1		
	signal_to_clutter_ratio	int		db	0.1		
	op_command	Complex Type					
	op_response	Complex Type					
	engagement_result	enum	EngagementResultEnum				

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	miss_distance_x	long		m	0.1		
	miss_distance_y	long		m	0.1		
	miss_distance_z	long		m	0.1		
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	track_status	bool					
	checksum	unsigned long					
MWRadarPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	local_target_tag	int					
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	platform_serial_number	int					
	boresight_azimuth	int		deg	0.1		
	boresight_elevation	int		deg	0.1		
	azimuth_valid	bool					
	elevation_valid	bool					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	range_valid	bool					
	doppler_valid	bool					
	injection_verified	bool					
	emission_verified	bool					
	checksum	unsigned long					
CommandGuidedMissilePDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	radar_serial_number	int					
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_response	Complex Type					
	armed	bool					
	launch_status	bool					
	target_lock	bool					
	checksum	unsigned long					
ActiveMissilePDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	radar_serial_number	int					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_response	Complex Type					
	armed	bool					
	launch_status	bool					
	target_lock	bool					
	checksum	unsigned long					
ArtilleryPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	radar_serial_number	int					
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_response	Complex Type					
	armed	bool					
	launch_status	bool					
	target_lock	bool					
	checksum	unsigned long					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
SemiActivePDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	radar_serial_number	int					
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_response	Complex Type					
	armed	bool					
	launch_status	bool					
	target_lock	bool					
	checksum	unsigned long					
TestDirectorPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	director_command	Complex Type					
	director_response	Complex Type					
	checksum	unsigned long					
FacilityControllerPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	director_command	Complex Type					
	director_response	Complex Type					
	checksum	unsigned long					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
RadarWarningPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	platform_serial_number	int					
	warning	bool					
	emission_verified	bool					
	checksum	unsigned long					
ChaffDispenserPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	platform_serial_number	int					
	op_command	Complex Type					
	op_response	Complex Type					
	checksum	unsigned long					
JammerPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	power_percent	int		%	0.1		
	platform_serial_number	int					
	op_command	Complex Type					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	op_response	Complex Type					
	injection_verified	bool					
	emission_verified	bool					
	checksum	unsigned long					
TowedJammerPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	mode_serial_number	int					
	power_percent	int		%	0.1		
	platform_serial_number	int					
	position_offset_x	long		m	0.1		
	position_offset_y	long		m	0.1		
	position_offset_z	long		m	0.1		
	op_command	Complex Type					
	op_response	Complex Type					
	injection_verified	bool					
	emission_verified	bool					
	checksum	unsigned long					
C3PDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	op_command	Complex Type					
	op_response	Complex Type					
	checksum	unsigned long					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
StandAloneEnvironmentMonitorPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	verified	bool					
	checksum	unsigned long					
CommunicationECMPDUStruct	SerialNumber	int					
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	azimuth	int		deg	0.1		
	elevation	int		deg	0.1		
	range	long		m	0.1		
	doppler	long		Hz	1		
	op_command	Complex Type					
	op_response	Complex Type					
	engagement_result_enum	enum					
	azimuth_valid	bool					
	elevation_valid	bool					
	range_valid	bool					
	doppler_valid	bool					
	armed	bool					
	launch_status	bool					
	target_lock	bool					
	checksum	unsigned long					
PlatformPDUStruct	SerialNumber	int					

Complex Datatype Table							
Complex Datatype	**Field**	**Datatype**	**Cardinality**	**Units**	**Resolution**	**Accuracy**	**Accuracy Condtion**
	time	unsigned long		ms	1		
	entity_status	enum	EntityStatusEnum				
	communication_status	enum	CommunicationStatusEnum				
	position_x	long		m	0.1		
	position_y	long		m	0.1		
	position_z	long		m	0.1		
	orientation_phi	int		deg	0.1		
	orientation_theta	int		deg	0.1		
	orientation_psi	int		deg	0.1		
	op_command	Complex Type					
	op_response	Complex Type					
	checksum	unsigned long					
OpCommandStruct	op_code	enum	OpCodeEnum				
	data	char[34]					
	checksum	unsigned long					
OpResponseStruct	op_code	enum	OpCodeEnum				
	data	char[34]					
	checksum	unsigned long					
DirCommandStruct	op_code	enum	OpCodeEnum				
	data	char[16]					
	checksum	unsigned long					
DirResponseStruct	op_code	enum	OpCodeEnum				
	data	char[16]					
	checksum	unsigned long					

APPENDIX D. TSLA ENUMERATED DATA TYPES

Enumerated Datatype Table		
Identifier	**Enumerator**	**Representation**
CategoryEnum	CATEGORY_UNKNOWN	0
CommunicationStatusEnum	COMM_OKAY COMM_LATE COMM_FAULT	0 1 2
CountryEnum	COUNTRY_UNKNOWN	0
DirOpCodeEnum	DIR_NOP	0
DomainEnum	DOMAIN_UNKNOWN	0
ECMTechniqueEnum	ECM_UNKNOWN ECM_NOISE ECM_FALSE_TARGET	0 1 2
EngagementResultEnum	ENGAGE_UNKNOWN ENGAGE_PENDING ENGAGE_ENGAGING ENGAGE_KILL ENGAGE_NO_KILL	0 1 2 3 4
EngagementStatusEnum	ENGAGE_NOT_ENGAGED ENGAGE_ENGAGE	0 1
EntityKindEnum	KIND_UNKNOWN	0
EntityStatusEnum	ENTITY_ON ENTITY_OFF ENTITY_STANDBY ENTITY_FAIL	0 1 2 3
ExtraEnum	EXTRA_UNKNOWN	0
OpCodeEnum	OP_NOP OP_ARM OP_ASSIGN OP_DETONATE OP_GUIDE OP_HANDOFF OP_INTERCEPT OP_LAUNCH OP_SWITCH OP_SENDLOCKSTATUS	0 1 2 3 4 5 6 7 8 9

Enumerated Datatype Table		
Identifier	**Enumerator**	**Representation**
	OP_SENDTARGETZONE OP_CHAFF_DISPENSE	10 11
RadarTypeEnum	RADAR_UNKNOWN RADAR_EW RADAR_HF RADAR_TAR RADAR_TTR RADAR_AIR RADAR_MWR	0 1 2 3 4 5 6
RadarOperationEnum	RADAROP_UNKNOWN RADAROP_SCAN RADAROP_TRACK RADAROP_HEIGHTFIND	0 1 2 3
SubCategoryEnum	SUBCATEGORY_UNKNOWN	0
SpecificEnum	SPECIFIC_UNKNOWN	0

APPENDIX E. LATENCY COMPENSATION METHOD

E.1. PERFORMANCE REQUIREMENTS

The distributed nature of the test network, and the processing time of the RTI will insert some amount of latency in the delivery of data to the end user. For the purposes of this specification, latency is defined as the difference in time between when data are measured by the data generator and when the data reaches the data consumer. The data generator shall time tag the data to correspond to the time that the measurement was made. The data consumer shall record the time the data was provided. The time difference is the latency.

The approach to latency compensation is to estimate the *state* of the dynamic system (e.g. the platform position and velocity in a 6-state model, or its position, velocity and acceleration in a 9-state model) and then use these estimates to predict its future position. If the state estimates are accurate when transmitted, then predictions based on a kinematic extrapolation will remain accurate until the highest derivative of the target state that is used in the predictor changes from the value in the transmitted estimate.

E.1.1. LATENCY RANGE

The expected range of values for latency in the test network is difficult to determine reliably. Also, it is recognized that latency compensation can only be performed for those data sources where the highest derivatives in the underlying system model remain constant during the interval of compensation. (The state variables include things like *position* which need not remain constant (of course). What must remain constant is either the system model, or perhaps the highest derivative modeled). For flight encounters of interest, it would be unreasonable to expect that the highest order derivative would remain constant for longer than 0.5 seconds. Hence, on this basis, the maximum range of compensation for latency shall be 0.5 seconds.

E.1.2. PREDICTION ACCURACY

The accuracy of prediction following latency compensation shall be less than 0.5 meters for range and less than 0.1 milliradian for azimuth and elevation.

E.2. DATA UPDATE INTERVAL

The required update interval is dictated by a) the actual data transmitted from generator to consumer, and b) the minimum expected interval of time over which the highest derivatives of the state variables can be expected to remain constant. For the platforms of interest to EW testing (i.e., tactical aircraft and missiles) the maximum interval over which these state variables can be expected to remain constant is 25ms. This shall be the maximum update interval for state estimates.

E.3. COMPUTATION OF STATE ESTIMATES

The TSLA network shall include, at each data generator, a filter capable of producing estimates of the current (3-space) position of the platform, and the first time derivatives of these parameters. The updates of these estimates shall be provided at a 40Hz rate. The TSLA network shall require that a predictor algorithm be implemented at each data consumer (where latency compensation is required) which will operate with the state estimates to produce predictions with accuracy as specified in Section E.1. If there is a requirement for raw (unfiltered data) these data values shall be transmitted within the same data structure as that for the filtered state estimates.

The rationale for this specification is provided in the paragraphs below.

There are two parts to the latency compensation process. The first part is a filtering process which produces accurate estimates of the system state variables. The second part is an extrapolation process which uses the state estimates to form a predicted value of position based on the state estimates. The prediction computation is relative to the time of the state estimates. The time for the prediction is the current time.

There are three possible implementations of the filter-predictor process. In the first two methods, the filter is located at the data generator, while predictor is resident at the data consumer. There are two cases here because, with the filter, it is possible to provide updates of the state estimates at a rate different from the sample interval provided by the measurement sensor. The data generator can monitor test the results of predictions which would be made by the current state estimates, and only provide updates to the state estimates when these predictions exceeded some error tolerance. Or, in the second approach, the data generator could simply provide a state update each time there was a new measurement. The first approach provides the opportunity to reduce the amount of data traffic. But, it requires the data generator to test for the condition when the prediction error becomes too large. Also, this approach has the limitation that the state is updated only after it is determined that the current state estimate will produce excessive prediction error. Hence, there is an inherent latency built in to this approach.

The third approach places both the filter and the predictor at the data consumer node. This approach is appealing whenever there is a need for the raw sensor data to be transmitted over the network (For example if it is required for use in test data reduction.) In this situation, both the filter and the predictor are implemented at the data consumer node. Here, the drawback is that each data consumer must now support the computational burden of the filter.